Universidad Nacional Mayor de San Marcos
Universidad del Perú. Decana de América

Dirección General de Estudios de Posgrado
Facultad de Ciencias Matemáticas
Unidad de Posgrado

K − vecino más próximos en una aplicación de clasificación y predicción en el Poder Judicial del Perú

TESIS

Para optar el Grado Académico de Magíster en Estadística Matemática

AUTOR

Nel QUEZADA LUCIO

ASESOR

Wilfredo Eugenio DOMÍNGUEZ CIRILO

Lima, Perú

2017

Referencia bibliográfica

Quezada, N. (2017). *K − vecino más próximos en una aplicación de clasificación y predicción en el Poder Judicial del Perú*. [Tesis de maestría, Universidad Nacional Mayor de San Marcos, Facultad de Ciencias Matemáticas, Unidad de Posgrado]. Repositorio institucional Cybertesis UNMSM.

590

15(R)

124

ACTA DE SUSTENTACIÓN DE TESIS DE GRADO ACADÉMICO DE MAGÍSTER

Siendo las, 16:20 horas del día martes veintiuno de febrero del dos mil diecisiete, en el Auditorio de la Facultad de Ciencias Matemáticas, el Jurado Evaluador de Tesis, Presidido por la Mg. Ana María Cárdenas Rojas e integrado por los siguientes miembros, Mg. Rosa Ysabel Adriazola Cruz (Jurado Evaluador); Mg. Rosa Fátima Medina Merino (Jurado Evaluador), Dra. Ilse Janine Villavicencio Ramírez (Jurado Evaluador) y el Mg.Wilfredo Eugenio Domínguez Cirilo como Miembro Asesor, se reunieron para la sustentación de la tesis titulada: «K-VECINO MÁS PROXIMOS EN UNA APLICACIÓN DE CLASIFICACIÓN Y PREDICCIÓN EN EL PODER JUDICIAL DEL PERÚ» presentada por el Bachiller Nel Quezada Lucio, para optar el Grado Académico de Magíster en Estadística Matemática.

Luego de la exposición del graduando, los Miembros del Jurado hicieron las preguntas correspondientes, así como las observaciones e inquietudes acerca del trabajo de tesis, a las cuales el Bachiller Nel Quezada Lucio respondió con acierto y solvencia, demostrando pleno conocimiento del tema.

A continuación se realizó la calificación correspondiente, según tabla adjunta, resultando el Bachiller Nel Quezada Lucio aprobado con el calificativo de ..16..... BUENO.....

Habiendo sido aprobada la sustentación de la Tesis, el Jurado Evaluador recomienda para que el Consejo de Facultad apruebe el otorgamiento del grado académico de **Magíster en Estadística Matemática** al Bachiller Nel Quezada Lucio.

Siendo las 17:30 horas, se levantó la sesión, firmando para constancia la presente Acta.

Rosa Ysabel Adriazola Cruz
Miembro

Mg. Ana María Cárdenas Rojas
Presidenta

Mg. Rosa Fátima Medina Merino
Miembro

Dra. Ilse Janine Villavicencio Ramírez
Miembro

Mg. Wilfredo ~~Eugenio~~ Domínguez Cirilo
Miembro Asesor

Dedicatoria

A Francisca Lucio Azaña, por ser una madre extraordinaria.

ÍNDICE GENERAL

Lista de Cuadros

Lista de Figuras

Resumen

El presente trabajo de investigación se fundamenta en la construcción de modelos utilizando el método de los k−vecinos más próximos con el propósito de clasificar y predecir a las Cortes Superiores de Justicia del Poder Judicial del Perú. Con la intención de identificar y evaluar a las 31 Cortes Superiores de Justicia, se realiza un análisis descriptivo de datos, el cual permite realizar una estimación *a priori* del número de conglomerados y localizar los valores atípicos cada una de las variables, apoyado en estas estadísticas la Corte de Lima es excluida del estudio. Con las 30 Cortes Superiores, se encuentra un modelo de agrupamiento *a posteriori* el cual forma tres grupos (pequeño, mediano y grande), fundado en conglomerados jerárquicos (clasificación no supervisada), para ello se calcula la matriz de distancia euclidiana, apoyado en el método del vecino más próximo se encuentra el grado jerarquía de cada Corte Superior, que permite construir el árbol de clasificación (dendrograma) que representa al modelo de agrupamiento *a posteriori*. En seguida mediante el método (algoritmo) de los k-vecinos más próximos (clasificación supervisada) se encuentra el modelo de clasificación para ello se realiza la estimación *a priori* del valor de k, luego se define la partición de entrenamiento y reserva para validar el modelo, para mejorar el valor de k se realizan pliegues para la validación cruzada y se desarrolla el algoritmo de los 3-vecinos más próximos (3-NN), que encuentra el modelo llamado espacio de predictores, el cual ubica al caso focal y sus vecinos en el gráficos de homólogas. De otro lado el modelo encontrado evidencia la importancia de las variables (predictores), valora las distancias en la tabla de vecinos y distancias y analiza los promedios con los mapas de cuadrantes. La significancia del modelo se evidencia en el resumen de errores de la tabla de clasificación y el gráfico de registros de errores de selección del valor a posteriori de k. Además se muestra la clasificación pronosticada, las probabilidades de clasificación mediante 3 - vecinos más próximos. Se demuestra que el modelo encontrado se ejecuta con precisión para pequeños tamaños de muestra de entrenamiento y reserva, mediante las pruebas no paramétricas de Kruskal-Wallis y la

Mediana, en ambas pruebas rechazamos la hipótesis de igualdad de promedios y medianas poblacionales respectivamente, y concluimos que los grupos (pequeña, mediana y grande) comparados difieren en cada una de las seis variables. Finalmente se muestra un modelo predictivo (algoritmo) para clasificar futuras Corte Superiores de Justicia en el Poder Judicial, que permite se realiza una simulación de predicción utilizando las variables de la Corte Superior de Lima.

Palabra Clave: Conglomerados, Clasificación supervisada y no supervisada, k-Vecinos más próximos, Clasificación no paramétrica, Distancia Euclidiana. Poder Judicial, Cortes Superiores de Justicia. Validación cruzada para muestras pequeñas.

Abstract

The present research work is based on the construction of models using the method of the nearest k-neighbors with the purpose of classifying and predicting the High Courts of Justice of the Judicial Power of Peru. With the intention of identifying and evaluating the 31 Supreme Courts of Justice, a descriptive analysis of data is carried out, which allows to make an a priori estimate of the number of clusters and to locate the atypical values of each of the variables, supported by these statistics The Court of Lima is excluded from the study. With the 30 Upper Courts, a posterior clustering model is found which forms three groups (small, medium and large), based on hierarchical clusters (unsupervised classification), for which the Euclidean distance matrix, Method of the nearest neighbor is the degree hierarchy of each Superior Court, which allows to construct the classification tree (dendrogram) that represents the model of a posteriori grouping. Next, by means of the method (algorithm) of the nearest k-neighbors (supervised classification), the classification model is found for this, the a priori estimation of the value of k is made, then the training and reserve partition is defined to validate the Model, to improve the value of k folds are made for cross-validation and the algorithm of the closest 3-neighbors (3-NN) is developed, which finds the model called space of predictors, which locates the focal case and its Neighbors in the graph of homologues. On the other hand the model found evidences the importance of the variables (predictors), evaluates the distances in the table of neighbors and distances and analyzes the averages with the maps of quadrants. The significance of the model is evidenced in the summary of errors of the classification table and the graph of error records of selection of the posterior value of k. In addition, the predicted classification is shown, the probabilities of classification by 3-nearest neighbors. It is demonstrated that the model found is executed accurately for small sample sizes of training and reserve, using non-parametric Kruskal-Wallis and Median tests, in both tests we reject the assumption of equality of averages and population medians respectively, and we conclude That the groups (small, medium and large) compared differ in each of the six variables.

Finally, a predictive model (algorithm) is presented to classify future Superior Courts of Justice in the Judiciary, which allows a simulation of prediction using the variables of the Superior Court of Lima.

Key word: Conglomerates, Supervised and unsupervised classification, k-Nearest neighbors, Non-parametric classification, Euclidean distance. Judicial Branch, Superior Courts of Justice. Cross-validation for small samples.

I. INTRODUCCIÓN

1.1. Situación Problemática

En el Poder Judicial del Perú, es urgente mejorar la gestión administrativa, concerniente a la distribución de los recursos económicos, logísticos, humanos y ajustar los estándares de producción de las dependencias jurisdiccionales. Además, es frecuente enfrentarse con la necesidad de identificar las principales características de las Cortes Superiores de Justicia. Con la finalidad de poder optimizar los recursos y evaluar mejor la producción jurisdiccional. Es importante encontrar un modelo para clasificar a las 31 Cortes Superiores de Justicia y poder realizar predicciones para futuras Cortes Superiores. También es necesario que el modelo de clasificación encontrado sea ideal, precisó, y revelador que permita solucionar estas carencias a la hora de otorgar los recursos económicos, logísticos, humanos y evaluar la productividad jurisdiccional. El objetivo es encontrar un modelo que permita clasificar las 31 Cortes Superiores de Justicia y predecir futuras Cortes Superiores utilizando el m*étodo no paramétrico* llamado "K-*Vecinos más próximos", y* generar un interés natural en la comunidad judicial respecto a los beneficios que se obtendrán al aplicar estos métodos estadísticos. Además se debe indicar que la viabilidad de la aplicación del modelo es posible dado que se tiene acceso a los datos y se conoce las principales tipologías de las actividades que se desarrollan en la institución.

1.2. Formulación del Problema

En el Poder Judicial existe inconveniente a la hora de la distribución de recursos económicos, logísticos, humanos y ajustar los estándares de productividad de las dependencias jurisdiccionales, debido a que se vienen considerando a todas las Cortes Superiores de Justicia como si tuvieran las mismas características, Es decir, todas las Cortes Superiores

Son iguales en la distribución de los recursos y los estándar de producción. Con el fin de poder optimizar los recursos y evaluar mejor la productividad es importante encontrar un modelo que solucione estos inconvenientes.

1.2.1 Problema General

- ¿Cómo desarrollar modelos mediante el método de los k−vecinos más próximos que permita clasificar y predecir a las 31 Cortes Superiores de Justicia del País?

1.2.2 Problemas Específicos

- ¿Cómo implantar un modelo para los predictores (variables) basado en el método de los k vecinos más próximos para clasificar y predecir las Cortes Superiores de Justicia?

- ¿Cómo implementar un modelo de k vecinos más próximos cuando se tiene muestras pequeñas de entrenamiento y reserva (validación)?

- ¿Cómo establecer modelos que permitan identificar y evaluar a las 31 Cortes Superiores de Justicia, respecto de los predictores (variables) en forma a priori?

- ¿Cómo desarrollar un modelo jerárquico mediante el método de encadenamiento simple (vecinos más próximos) para agrupar las Cortes Superiores de Justicia en conglomerados?

1.3. Justificación de la Investigación

Con la finalidad de mejorar la gestión administrativa en el Poder Judicial del Perú, existe la necesidad apremiante de identificar grupos de Cortes Superiores de Justicia con características comunes. De otro lado es importante encontrar un modelo que nos permitirá clasificar las Cortes Superiores de Justicia en uno de los grupos y poder realizar predicciones para futuras Cortes.

En consecuencia mediante conglomerados jerárquicos utilizando el método de encadenamiento simple (vecinos más próximos) nos permitirá identificar y diferenciar los grupos de Cortes Superiores de Justicia basados en sus características que presentan cada una de ellas.

De otro la mediante la aplicación del método k vecinos más próximos permitirá establecer un modelo que sea capaz de clasificar con precisión a las Cortes Superiores de Justicia en uno de los grupos, también este modelo debe permitir predecir la clasificación de futuras Cortes Superiores de Justicia, además se debe mostrar las variables más importantes en las estimaciones realizadas.

El presente estudio se realiza con 31 Cortes Superiores de Justicia del país y seis variables en estudio que son: Pendientes (Cantidad de procesos que provienen de un inventario físico), Ingresos (cantidad de procesos principales equivalentes a al incremento de la carga procesal, Resueltos (Cantidad de procesos principales que implican la disminución de la carga procesal en el mes), Población (Cantidad de habitantes en cada Corte Superior de Justicia del País), Órganos jurisdiccionales (Cantidad de dependencias jurisdiccionales en cada Corte Superior de Justicia del País). Personal (Cantidad de personas que laboral en cada Corte Superior de Justicia del País).

1.3.1. Justificación Práctica

Mediante el modelo de clasificación y predicción aplicada a las Cortes Superiores Justicia del país permitirá mejorar la distribución de recursos económicos, logísticos, humanos y evaluar la productividad de las dependencias jurisdiccionales del Poder Judicial, en cada uno de los conglomerado (grupo) formados, el modelo de clasificación y predicción permitirá actualizar los grupos de las Cortes Superiores de Justicia de acuerdo al estado actual de sus variables, permitiendo una mejora continua de los grupos formados. También se resolverá el problema de las variables más importante en las evaluaciones de la información. La solución de todos estos problemas será posible utilizando los métodos de clasificación de los k-vecinos más próximos.

1.3.2. Justificación Teórica

Mediante la aplicación del método k−vecinos más próximos se encuentra un modelo de clasificación y predicción en el Poder Judicial con la finalidad de buscar crear reflexión sobre los beneficios e importancia del desarrollo de estos métodos en la solución de problemas a la hora de la distribución de recursos económicos, humanos logísticos, así como en las estimaciones de algunos parámetros útiles para mejorar la gestión administrativa en la institución, de otro lado es importante generar un debate académico en la institución sobre lo conveniente del desarrollo y aplicación de estos métodos de clasificación no paramétrica. También es importante indicar que el método de k vecinos más próximos es preciso para modelos cuando el tamaño de la muestra de reserva y la muestra de entrenamiento son pequeños siempre cuando se aplique validación cruzada a los datos del estudio.

1.4. Objetivos de la Investigación

1.4.1 Objetivo General

- Encontrar modelos utilizando el método de los k−vecinos más próximos con el propósito de clasificar las 31 Cortes Superiores de Justicia del País y poder realizar predicciones para futuras Cortes Superiores de Justicia.

1.4.2 Objetivo Específicos

- Verificar la validez del modelo de clasificación y predicción de las Cortes Superiores de Justicia basado en el método de los k vecinos más próximos.

- Verificar la precisión del modelo de k vecinos más próximo cuando se tiene muestras pequeñas de entrenamiento y reserva (validación)

- Experimentar los modelos que identifican y evalúan a las 31 Cortes Superiores de Justicia, respecto de los predictores (variables) en forma a priori.

- Encontrar un modelo de agrupamiento jerárquico basado en encadenamiento simple (vecinos más próximos) para asociar las Cortes Superiores de Justicia del País en conglomerados.

II. MARCO TEÓRICO

2.1. Antecedentes de investigación

- **Huamanchumo de la Cuba, Luis (2005). "Estandarización de la Carga Procesal del Poder Judicial de la República del Perú: un enfoque factorial estructurado". TECNIA. Vol. 15, 1. Pp. 41-49. ISSN N°0375-7765.**

  Encuentra evidencia empírica de la existencia de estándares en los juzgados de familia utilizando un modelo factorial estructurado basado en un supuesto de correlación estadística entre la carga procesal y la cantidad de expedientes resueltos en cada dependencia judicial. Propone que los estándares de carga procesal y producción judicial es el resultado de un proceso Input/Output (I/O), el cual, está determinado por la dotación de tecnología disponible y por los recursos humanos y materiales con los que cuenta la dependencia. Con la información estadística de la Gerencia General del Poder Judicial del año 2003 ha sido posible demostrar que los factores que explican los estándares de carga procesal en los juzgados civiles, laborales y de familia pueden resumirse en los factores Ubicación de la dependencia judicial (Sede de Corte/Fuera de Sede), y/o del Sistema Organizativo al que pertenecen De esta forma, mediante la aplicación de la técnica MANOVA estructurada se obtuvo los estándares observados en las mencionadas dependencias.

- **Salas Arenas, Jorge L. (2010). Bases para la racionalización de la carga procesal: justicia en el reparto de la tarea de administrar justicia.**

  Para efectos estadísticos en la actualidad, todas todos los procesos, tienen el mismo valor o peso, y por ello se presentan interna (Poder

Judicial) y exteriormente (colectividad) resultados anualizados de la producción jurisdiccional en número de causas ingresadas y resueltas, para estimar el nivel de respuesta de los diversos órganos jurisdiccionales a la demanda de justicia. Con aquella ponderación cuantitativa que se funda en la ficción de que todas las causas son simples; no resulta realmente reflejado el esfuerzo ni el costo de operación del sistema de justicia; en el camino se desprecia todo el esfuerzo y el costo de trámite que las causas complejas representa.

El estudio muestra, La Cuota Ideal, Tabla de Nomenclaturas, Impacto de la Carga Procesal en la Calidad de los Productos Jurisdiccionales, Productividad, Tiempo Razonable y Control de Plazos. El autor considera que una justa distribución de causas requiere estandarizar el proceso mediante el cálculo de un ponderador que incorpore su grado de complejidad.

- **Hernández Breña, Wilson (2007). 13 mitos sobre la carga procesal. Anotaciones y datos para la política judicial pendiente en la materia. Justicia Viva. Lima, Perú**.

Se da conocer sobre los mitos de la carga procesal del Poder Judicial, es decir, muestra conocimiento sobre cómo funciona la justicia que está rodeado de mitos que suenan bonito y hasta nos pueden hacer quedar bien. Sin embargo, insidiosamente empañan los caminos de soluciones efectivas. Porque hemos sido muchas veces víctimas de los mitos y seguramente de otros más, advertimos la necesidad de estudiarlos con serenidad y seriedad y brindar, sin apasionamientos, una opinión informada y objetiva sobre la realidad de la carga procesal.

El objetivo es contribuir con información para conocer mejor lo que ya conocemos y criticamos, ampliar el campo de diagnóstico, dejar de

responsabilizar siempre al juez, promover consensos en torno de temas cotidianos, generar información útil para la toma de decisiones y plantear mejor las soluciones. Este documento consta de 13 apartados, cada uno elaborado alrededor de un mito sobre la carga procesal, Cada apartado incluye argumentos cuantitativos y cualitativos que llevan las conclusiones hacia su desmitificación como verdades sagradas. Finalmente concluye que la falta de una dotación suficiente y adecuada tanto de la infraestructura como del equipamiento es evidente para lo cual una óptima distribución de recursos económicos, humanos y logísticos hace necesario trabajar con estándares.

- **Estrategia de regresión basada en el método de los k vecinos más próximos para la estimación de la distancia de falla en sistemas radiales**. **Germán Morales España, Juan Mora Flórez, Herman Vargas Torres (2008) Rev. Fac. Ing. Univ. Antioquia N.° 45 pp. 100-108. Septiembre, 2008.**

  Se presenta una estrategia de regresión para estimación de la distancia de falla en sistemas de potencia radiales, empleando la técnica de los k-Vecinos más próximos (k-NN). Esta propuesta de localización de fallas utiliza las medidas de la componente fundamental de tensión y de corriente disponibles en la subestación, no depende del modelo del sistema de potencia y se adapta a las características particulares de los sistemas radiales. La continuidad y por tanto la calidad del servicio de energía eléctrica se ve afectada por las fallas. Algunas técnicas relevantes aplicables a sistemas de radiales han sido planteadas para la localización de fallas. Utilizando la componente fundamental de la corriente y tensión en pre falla y falla medidas en la subestación, estiman la sección de línea fallada con la comparación de la impedancia obtenida a partir del modelo impuesto por el método y la impedancia equivalente calculada.

Se propone el uso de la técnica de aprendizaje supervisado de regresión, conocida como los k vecinos más próximos (k-NN). Esta técnica se aplica a la estimación de la distancia de falla, considerando las características fundamentales de los sistemas radiales, sin depender del modelo del sistema. Inicialmente se presentan los fundamentos básicos de la técnica del vecino más próximo aplicada a la regresión. Luego se discute la aplicación de los k-NN a la localización de fallas.

- **Clasificadores eficaces basados en algoritmos rápidos de búsqueda del vecino más próximo en Reconocimiento de Patrones. Francisco Moreno Seco (2004).**

La clasificación no paramétricos más simples es el clasificador basado en la regla del vecino más próximo, que consiste en clasificar el objeto desconocido en la clase de su vecino más próximo según la disimilitud o distancia. Prácticamente todos estos algoritmos se pueden extender fácilmente para encontrar los k vecinos más próximos.

El objetivo consiste en el estudio de un algoritmo rápido de búsqueda del vecino más próximo, el LAESA, para obtener una disminución en la tasa de error sin coste adicional, utilizando para ello k vecinos; una vez abordado este objetivo, las ideas aplicadas a este algoritmo se han extendido a otros algoritmos con resultados similares e incluso mejores, y se han generalizado en una nueva regla de clasificación: la regla de los k vecinos más próximos. La idea básica de esta regla es utilizar para la clasificación los k candidatos más próximos de entre los seleccionados por el algoritmo de búsqueda. Una buena parte del trabajo de esta tesis se ha dedicado a explorar las posibilidades y el comportamiento de esta regla, que produce tasas de acierto en la

clasificación similares a las de la regla de los k vecinos más próximos pero con el coste computacional de la regla del vecino más próximo.

2.2. Bases Teóricas

En la presente tesis las bases teóricas que se utilizan se desarrollan a continuación.

2.2.1 Método de los K vecinos más próximos

Dado un conjunto de puntos $P=\{p_1, \ldots, p_n\}$ en un espacio métrico X con función de distancia ***d***, permitiendo algún reprocesamiento en P de manera eficiente, se desea responder a dos tipos de solicitudes:

Espacio métrico: es un conjunto que lleva asociada una función distancia, es decir, que esta función está definida sobre dicho conjunto, cumpliendo propiedades atribuidas a la distancia, de modo que para cualquier par de puntos del conjunto, estos están a una cierta distancia asignada por dicha función.

Vecino más próximo: localizar el punto en P más próximo a $q \in X$.

Rango: Dado un punto $q \in X$, y $r > 0$, tornar todos los puntos $p \in P$ que satisfagan $d(p,q) \leq r$.

Se buscan enfoques no paramétricos caracterizados por la ausencia de hipótesis a priori sobre la distribución condicional del espacio de definición. Puesto que la base está en el cálculo de distancias. El algoritmo más sencillo para la búsqueda del vecino más próximo es el conocido como fuerza bruta o exhaustivo, que calcula todas las distancias de un individuo a los individuos de la "muestra" y asigna al conjunto de vecinos más próximos, aquel cuya distancia sea mínima.

De otro lado se debe precisar que los métodos basados *"en vecindad"* son fundamentalmente dependientes de la distancia y en consecuencia poseen características propias de ésta como la

cercanía, la lejanía y la magnitud de longitud, entre otras. Los métodos basados en vecindad, además de servir para tareas de *clasificación,* también se usan para *agrupación* de datos. Existen dos grupos de métodos de vecindad, según la forma en que se realiza el aprendizaje. El grupo de los métodos retardados (o lazy) y los no retardados (o eager).

a. Métodos no retardados (o eager):

En los métodos no retardados se generaliza un solo modelo (también a partir de casos conocidos) para todos los nuevos datos que se desean clasificar, y éstos únicamente son tomados en cuenta como datos de entrenamiento cuando se vuelve a construir un nuevo modelo general. Entre estos métodos tenemos a los algoritmos de árboles de decisión y discriminantes lineales.

Ejemplo: Discriminantes lineales (Hernández Orallo, Ramírez Quintano, y Ferri Ramírez, 2004, p.426). Muestra la clasificación usando discriminantes lineales de Fisher. Se tiene de datos que se representan como puntos en un plano (Figura N° 1), los cuales hacen referencia a tres variedades de lirios: setosa (S), versicolor (E) y virginica (A). En el Figura N° 1 se muestran dos atributos (longitud y ancho de pétalo) y la clase (tipo de lirio). Los atributos “longitud de pétalo” y “ancho de pétalo” (en cms.) constituyen los ejes “*x*” y “*y*” del plano, respectivamente. Para empezar a construir un modelo general para clasificación de datos a partir de los casos iniciales que se tienen, se calculan puntos medios (centroides) en cada aglomeración de datos de cada clase (el que los datos parezcan estar agrupados en forma natural demuestra coherencia de los datos). Posteriormente se calcula la distancia en línea recta entre cada centroide y luego se divide el espacio trazando rectas perpendiculares a las líneas mencionadas anteriormente. Estas rectas perpendiculares deben cortar justo en el centro las líneas que unen los centroides. El modelo resultante es el mostrado en la Figura N° 2. Como se observa, el modelo general que se ha construido a partir de los datos consiste en

una división del espacio en regiones claramente delimitadas, las cuales servirán de criterio para la clasificación de nuevos datos. Por ejemplo, consideremos que se desea clasificar un lirio cuya longitud y ancho de pétalo es de 2 cms. De acuerdo al modelo mostrado en Figura N°2, es claro que el nuevo dato sería clasificado como “S” pues el punto coordenado conformado por los valores dados a los atributos se ubica dentro de la zona en la que hay una presencia mayoritaria de puntos S. Como vemos, para clasificar nuevos puntos no es estrictamente necesario volver a trazar nuevos modelos, pues éstos se pueden ir clasificando de acuerdo al modelo establecido inicialmente.

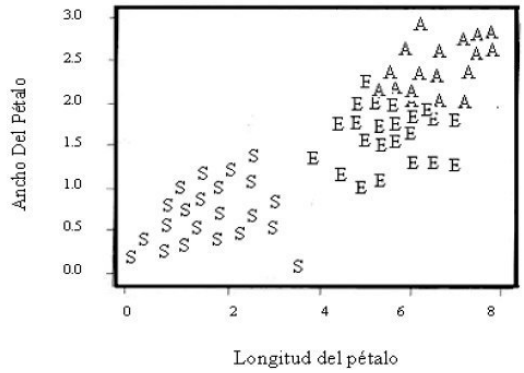


Figura 1: **Conjunto de datos para clasificar.** Rodríguez, J., Rojas E. & Franco, R. (2008)

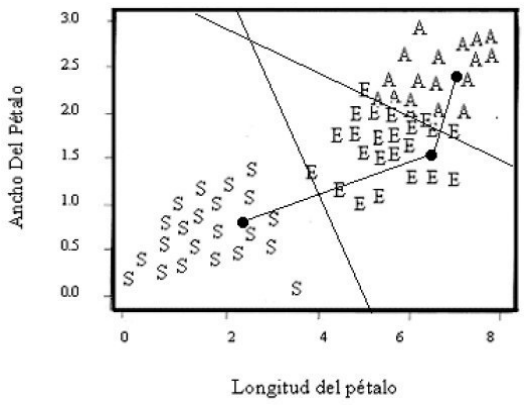


Figura 2: **Clasificación mediante discriminante de Fisher.** Rodríguez, J., Rojas E. & Franco, R. (2008)

b. Métodos retardados (o lazy):

En los métodos retardados como “k-vecinos-nn”, cada vez que se va a clasificar un dato, en la fase de entrenamiento, se elabora un modelo específico para cada nuevo dato, y una vez que éste se clasifica sirve como un nuevo caso de entrenamiento para clasificar una nueva instancia. (Técnicas Bayesianas).

Ejemplo: K-vecinos. Consideremos la misma situación de los lirios descrita en el ejemplo anterior. Para clasificar un nuevo dato con el método k-nn: primero, se ubica el dato a clasificar en el plano, supongamos que sus coordenadas son 7 y 2 de longitud y anchura respectivamente (Figura N° 3). Segundo, se determina un “radio de vecindad”. El valor de este radio puede ser asignado a partir de alguna heurística conocida. Tercero, se traza una circunferencia cuyo centro es el dato a clasificar; la circunferencia deberá encerrar uno o varios casos de entrenamiento próximos a la incógnita, si ninguno queda encerrado significan que la heurística usada para seleccionar el valor del radio no sirve y debe ser cambiada. Cuarto, se determina el valor de k (que también puede estar basado en una heurística), es decir, se establece si se va a comparar con el primer vecino más próximo, 1-nn, o con los dos vecinos más próximos, 2-nn; en fin, se le da un valor a k. Quinto, asignar la clase al nuevo elemento de acuerdo al valor de k y al número de datos encerrados en la circunferencia. En el gráfico 3 se han tomado dos radios diferentes con respecto al dato a clasificar. En la circunferencia con menor diámetro hay un individuo de la clase A y un individuo de la clase E. En este caso la clase que tomaría el nuevo dato sería la de la primera instancia más cercana del que supongamos es el A. Si el nuevo caso se quisiera clasificar con respecto a la circunferencia cuyo diámetro es mayor, el dato se clasificaría como A nuevamente, pues son mayoría dentro del vecindario seleccionado, ya que hay tres casos de la clase E y cinco de la clase A.

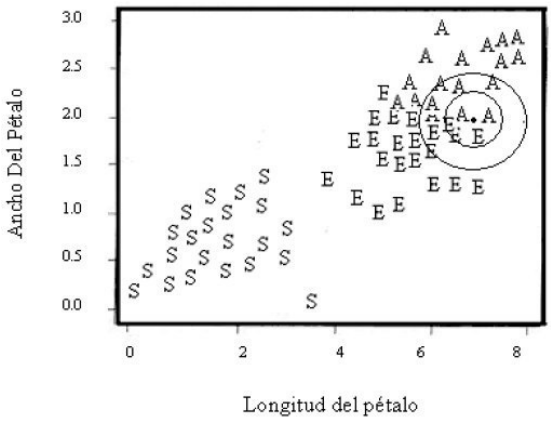


Figura 3: **Clasificación mediante Vecino más próximo.** Rodríguez, J., Rojas E. & Franco, R. (2008).

2.2.1.1. Métricas para medir distancia o similitud.

La similitud es una medida numérica que indica el grado al cual dos objetos se parecen. A más alto este valor más parecidos los objetos. Es no negativa y generalmente entre 1 (similitud máxima) y 0 (no hay similitud). Sin embargo, es común utilizar la distancia (inverso de la similitud), también conocida como disimilitud. Algunas medidas de similitud.

$$\text{Cos}(X, Y) = \frac{X \,.\, Y}{\|X\| \,.\, \|Y\|}$$

Ecuación 1: Para datos numéricos

Regla de decisión y selección de la distancia: El método de decisión está relacionado con la noción de proximidad o similitud entre los individuos. La distancia es el criterio de comparación principal usado en los métodos basados en Vecindad, por eso es conveniente mencionar algunas de las diferentes formas usadas para su medición.

Distancia de Minkowski: Es el índice de similitud más utilizado.

$$d(x,y) = [\sum_{i=1}^{n}(x_i - y_i)^p]^{\frac{1}{p}} \quad con \quad n \geq 1 \qquad (1)$$

Distancia Euclidea: Cuando *p=2* en (1)

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad (2)$$

Distancia de Manhattan: Cuando *p=1* en (1)

$$d(x, y) = \sum_{i=1}^{n} | x_i - y_i | \quad (3)$$

Distancia de Chebychev: Cuando $p \to \infty$ *1* en (1)

$$d(x, y) = \max_{i=1...n} | x_i - y_i | \quad (4)$$

Distancia de Mahalanobis:

$$d(x, y) = \sqrt{(x - y)^T S^{-1} (x - y)} \quad (5)$$

Distancia del Coseno:

$$d(x, y) = ar\cos\left(\frac{x^T y}{\| x \| \cdot \| y \|}\right) \quad (6)$$

Distancia usando la función delta: Sirve para hallar la distancia entre atributos nominales

$$d(x, y) = \omega \sum_{i=1}^{n} \delta(x_i, y_i) \quad (7)$$

Distancia entre dos conjuntos:

$$d(x, y) = \frac{| x \cup y | - | x \cap y |}{| x \cup y |} \quad (8)$$

Estas métricas satisfacen los requisitos matemáticos de una función de distancia.

- $d(x, y) \geq 0$
- $d(x, x) = 0$
- $d(x, y) = d(y, x)$
- $d(x, y) \leq d(x, h) + d(h, y)$

Distancia Ponderada: Se utiliza distancia ponderada para privilegiar a los vecinos más próximos y disminuir el papel de los vecinos lejanos, considerando las ideas de proximidad y lejanía relacionadas con la distancia d.

Sean (x_1,...,x_k) los k vecinos más próximos de x ordenados crecientemente por la distancia. A cada vecino se le asigna un peso w_i (S.A. Dudan, 1976):

$$w_i = \frac{d(x,x_k) - d(x,x_i)}{d(x,x_k) - d(x,x_1)} \quad si\ d(x,x_k) \neq d(x,x_1)$$

$$w_i = 1 \quad si\ d(x,x_k) = d(x,x_1)$$

El peso de los puntos debe variar inversamente con la distancia, de tal manera que los puntos más próximos tengan mayor peso. Se tiene entonces que:

$$w_i = K[\, d(x_r, x_d)]$$

K es la función Kernel que determina el peso de cada punto basado en la distancia al punto de referencia. Se pueden considerar las siguientes funciones:

$$K[d(x_d, x_r)] = \begin{cases} \dfrac{1}{d^2} \\ e^{-d} \\ \dfrac{1}{d_0 + d} \\ \dfrac{e^{-d(x_d, x_r)}}{\sum_{i=1}^{d} e^{-d(x_i, x_r)}} \end{cases}$$

En el caso de la última función, la asignación de pesos de esta forma tiene la característica de que $\sum w_i = 1$.

La selección del Kernel puede ser importante al momento de la imputación, por el valor (peso) que se le asignará a cada vecino. En la siguiente figura se muestra el comportamiento del Kernel en función de la distancia. Debido a la característica que debe cumplir el Kernel (debe variar inversamente con la distancia) su comportamiento es diferente si la distancia es mayor o menor que uno.

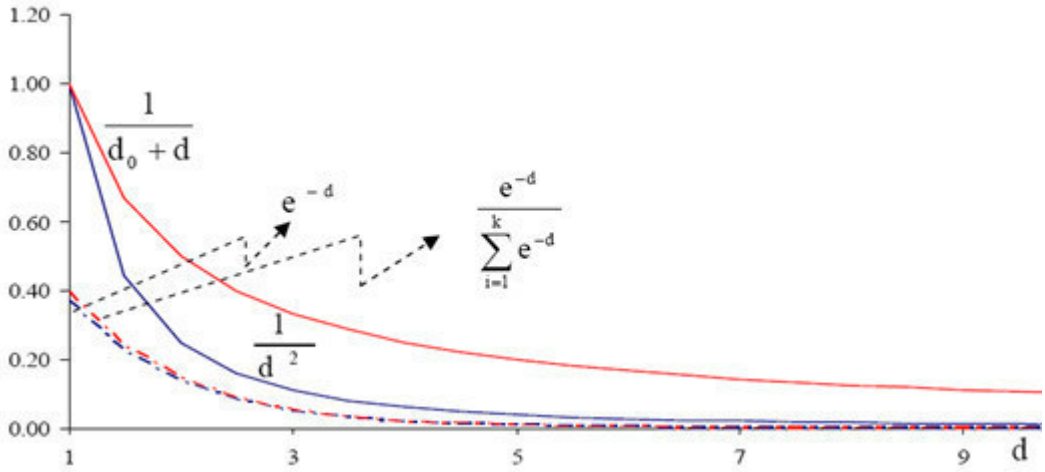


Figura 4: **Comportamiento del Kernel para distancias > 1.** Juárez, C.A. (2004).

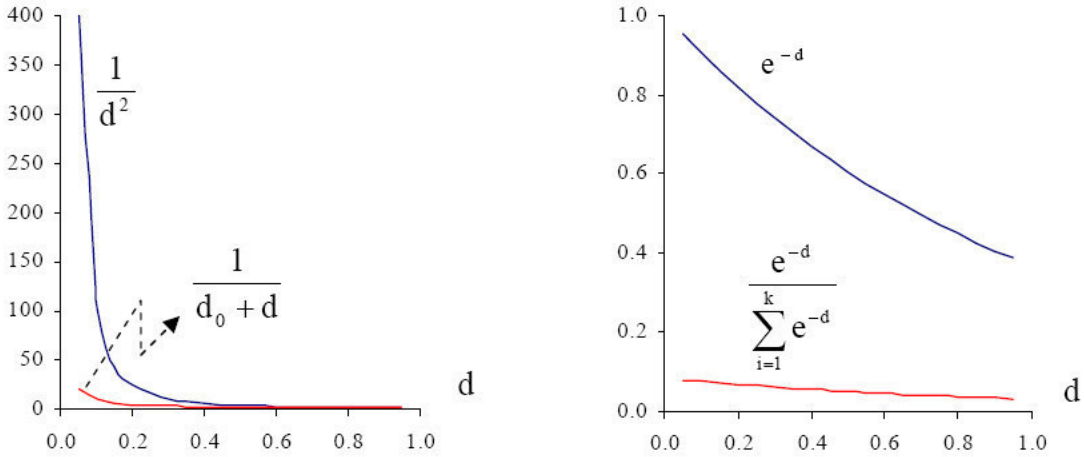


Figura 5: **Comportamiento del Kernel para distancias < 1.** Juárez, C.A. (2004).

Maldición de dimensionalidad (R. Bellman): Describe como aumenta la complejidad de un problema al aumentar la dimensión de las variables involucradas. Al aumentar la dimensión, el espacio está cada vez más vacío complicando cualquier proceso de inferencia. Al aumentar la dimensión aumenta su volumen y la densidad de la variable aleatoria disminuye.

En espacios con muchas dimensiones, el método del vecino más próximo puede presentar algunos problemas, debido a que la vecindad se hace muy grande. Por ejemplo, para el caso de 5000 puntos distribuidos uniformemente en un hípercubo unitario, se desea aplicar el método del vecino más próximo, considerando además que el punto de referencia está en el origen. En una dimensión, se recorrería en promedio una distancia de 5/5000 = 0.001 para tener a los 5 vecinos más próximos. En dos dimensiones, se recorrería en promedio una distancia de $\sqrt{0.001}$ para tener un cuadrado que tuviera 0.001 de volumen. En *d* dimensiones se recorrería $0.001^{1/d}$ (media geométrica).

Asimismo se ve afectado el método, si se incluyen características irrelevantes o ruidosas en los datos. La siguiente figura muestra dos puntos x_1 y x_2 próximos a la referencia ubicada en el origen.

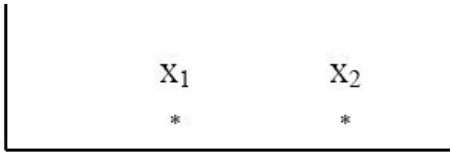


Figura 6: **Puntos en una dimensión < 1.** Juárez, C.A. (2004).

En una dimensión, el punto x_1 está más cerca de la referencia que x_2, sin embargo, si se le agrega una segunda característica aleatoria, los nuevos puntos más próximos podrían estar ubicados de otra forma.

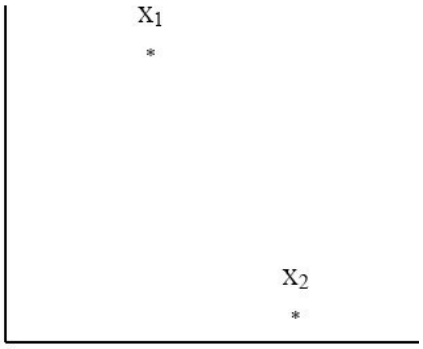


Figura 7: **Puntos en dos dimensiones < 1.** Juárez, C.A. (2004).

2.2.1.2. Métodos de búsqueda (algoritmos).

Existen muchos métodos, que proponen algoritmos eficientes para la búsqueda del vecino más próximo como son:

Algoritmos de aproximación y eliminación (Ramasubramanian y Paliwal, 2000)**:** Los más conocidos son el k-d tree (Bentley, 1975; Friedman et al., 1977), Fukunaga y Narendra (Fukunaga y Narendra, 1975), el vp-tree (Yianilos, 1993) y el GNAT (Brin, 1995), existen otros algoritmos de la familia AESA (Approximating Eliminating Search Algoritmo). El esquema general de búsqueda por aproximación y eliminación se resume:

1. Del conjunto, se selecciona un candidato a vecino más próximo.
2. Se calcula su distancia ***d*** al elemento en cuestión.
3. Si la distancia es menor que la distancia del vecino más próximo hasta el momento, d_{nn}, se actualiza el vecino más próximo y se eliminan del conjunto, aquellos individuos que no puedan estar dentro de una híperesfera de radio *d* y con centro en la muestra.
4. Se repiten los pasos anteriores hasta que no queden elementos por seleccionar en el conjunto.

Otra clasificación de los algoritmos rápidos de búsqueda de vecinos es la siguiente:

Búsqueda rápida local: Pone en evidencia una estructura sobre el conjunto elementos disminuyendo el número de distancias a calcular.

- Método de Friedman, Baskettand y Shusted
- Método de Yunk
- Método de Kittler

Estructura del conjunto de aprendizaje: Definí una zona restringida en donde se encontrarán los vecinos más próximos.

- Método de Fukunaga/Narendra
- Método de Delannoy

Reducción del conjunto de aprendizaje:

- Método de condensación de Hart
- Método de edición de Wilson

a).- K-Dimensional Tree (k-d tree)

K representa la dimensión de los datos del espacio de representación. Es un árbol binario que contiene en cada nodo intermedio información acerca de una coordenada que divide en dos el conjunto de datos del subárbol correspondiente al nodo, y en las hojas contiene buckets (puñados) de elementos. Para que el árbol resulte más equilibrado, se elige el híperplano de forma que se sitúe en la mediana de los valores de la coordenada discriminante. La coordenada discriminante debe ser aquella que tenga una mayor amplitud. Durante la fase de clasificación se recorre el árbol siguiendo un esquema de ramificación y poda para encontrar el vecino más próximo. El proceso de

búsqueda en el k-d tree es recursivo. Dado un elemento x, en un nodo (c) cualquiera del árbol se compara la coordenada de x que es discriminante, con el valor de corte (v) y se procede en la dirección más próxima según esa coordenada. Si $x(c)+d_{nn} \leq v$, el hijo derecho de ese nodo no puede contener al vecino más próximo y por tanto no es necesario buscarlo en ese nodo; de forma similar, si $x(c)-d_{nn} \geq v$, el hijo izquierdo no es necesario visitarlo. Si el nodo es una hoja, la muestra se compara con todos los individuos contenidos en ella.

b).- Vantage Point Tree (vp-tree)

Es un árbol binario en el que cada nodo representa un subconjunto S del conjunto y utiliza un elemento del conjunto llamado pivote para dividir el conjunto S en dos subconjuntos, uno por cada hijo. Cada nodo contiene dos hijos. El hijo izquierdo contiene el subárbol correspondiente al subconjunto de S que está a una distancia del pivote menor que un cierto valor μ y el hijo derecho contiene el subárbol correspondiente al resto de S (ambos subconjuntos de tamaño similar). En cada nodo se almacenan, junto con el pivote y el valor de μ, la cota inferior y superior de las distancias del pivote a los individuos de cada subconjunto. La búsqueda en el árbol se realiza con el esquema de ramificación y poda, utilizando las cotas almacenadas en cada nodo para dirigir la búsqueda.

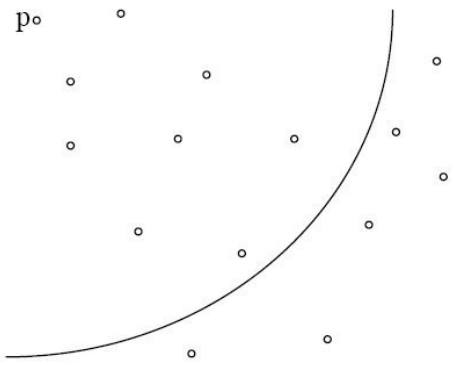


Figura 8: ***P*artición de un conjunto.** Juárez, C.A. (2004).

c).- Geometric Near-neighbour Access Tree (GNAT)

Este algoritmo se desarrolla de la siguiente manera:

1. Se eligen k puntos de partición, $p_1,p_2,\ldots,p_k$ del conjunto de datos, k (grado) puede variar en cada nodo del árbol. Se elige el primer punto al azar, el segundo es el más lejos del primero; el tercero es aquel cuya distancia al más próximo de los anteriores sea la mayor de los puntos restantes y así sucesivamente hasta elegir los k.
2. Divide el conjunto inicial en subconjuntos D_{pi}, cada uno asociado a un punto de partición p_i. Cada individuo p estará en el subconjunto asociado al punto de partición más próximo a él.
3. Para cada par de puntos de partición (p_i,p_j), se calcula y almacena el rango de p_i al conjunto D_{pj}, rango (pi,D_{pj}), que es un par de valores: el mínimo y el máximo de d(pi,p) para $p \in D_{pj} \cup \{p_j\}$, denotados como $min\_d(pi,D_{pj})$ y $max\_d(pi,D_{pj})$, respectivamente.
4. Construir recursivamente el árbol para cada Dpi, utilizando el mismo o diferente grado.

El algoritmo de búsqueda se formula de la siguiente manera:

1. Sea P el conjunto de puntos de partición de un nodo. Cada elemento de P se corresponde con un hijo del nodo en el árbol y representa un conjunto de puntos.
2. Elegir un individuo p de P (que no haya sido elegido) y calcular su distancia al individuo x, d(x,p). Se actualiza si es necesario el vecino más próximo hasta el momento y el rango de la búsqueda, r.
3. Para todo $q \in P$, $q \neq p$, si la intersección del rango [d(x,p)-r, d(x,p)+r] con rango (p,D_q) es vacía, eliminar q de P (utilizar desigualdad triangular). Se explica. Sea un individuo $y \in Dq$,
   a) Si $d(y,p) < d(x,p)-r$, como $d(x,y)+d(y, p) \geq d(x, p)$, entonces $d(x,y)>r$, luego y no puede ser el vecino más próximo.
   b) Si $d(y,p)>d(x,p)+ r$, como $d(y,x) + d(x,p) \geq d(y,p)$, entonces $d(x,y)>r$, luego y no puede ser el vecino más próximo.

4. Repetir los pasos 2 y 3 hasta que todos los individuos de P se hayan elegido o eliminado. Para los pi ∈ P no eliminados, repetir la búsqueda recursivamente en Dpi si el nodo de pi no es una hoja.

En la búsqueda el rango va cambiando (siempre es la distancia) y es un factor determinante en la eficacia de la búsqueda.

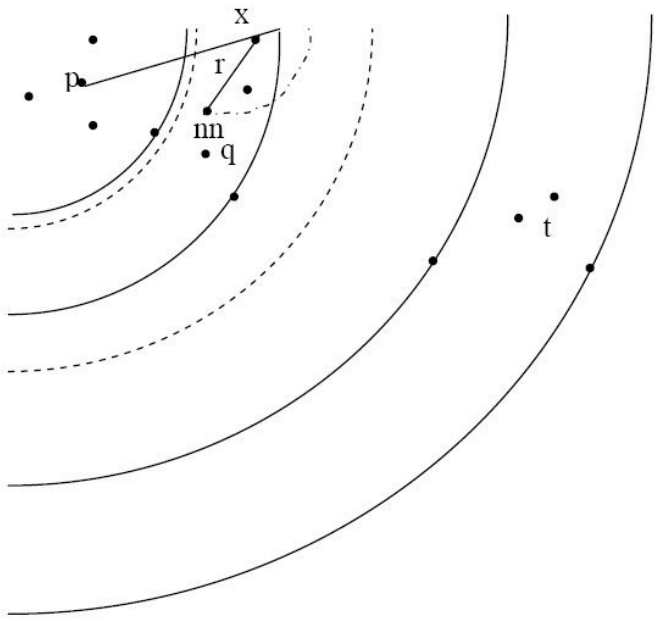


Figura 9: ***Eliminación en el algoritmo GNAT.*** Juárez, C.A. (2004).

Dado p y la distancia al vecino más próximo a x es r, se puede observar que la intersección del rango [d(x,p)-r, d(x,p)+r] (líneas punteadas) con rango (p,Dq) no es vacía, no podría eliminarse de forma segura el nodo de q. La intersección con rango (p,Dt) si resulta vacía, por lo que el nodo asociado a t puede eliminarse porque seguro no contiene al vecino más próximo.

d).- Algoritmo de Fukunaga / Narendra

Estructura el conjunto de elementos definiendo una zona restringida en donde se encontrarán los vecinos más próximos. El enfoque básico consiste en descomponer jerárquicamente las muestras en

subconjuntos disjuntos, y después aplicar a los grupos resultantes el método de Branch and Bound.

Se tiene:

$[X_1, X_2,..., X_N]$ N muestras n-dimensionales.

Se determinan los k-vecinos más próximos de una muestra X medidos por una función de distancia. El conjunto se divide en l subconjuntos, cada subconjunto es dividido a su vez en l subconjuntos y así sucesivamente. El resultado se representa por una estructura de árbol. Cada nodo p del árbol representa a un grupo de muestras, y está caracterizado por los siguientes parámetros:

- Sp conjunto de muestras asociadas al nodo p
- Np número de muestras asociadas al nodo p
- Mp media muestral de Sp
- rp MAX d(Xi,Mp) distancia más grande desde Mp a un Xi ∈ Sp

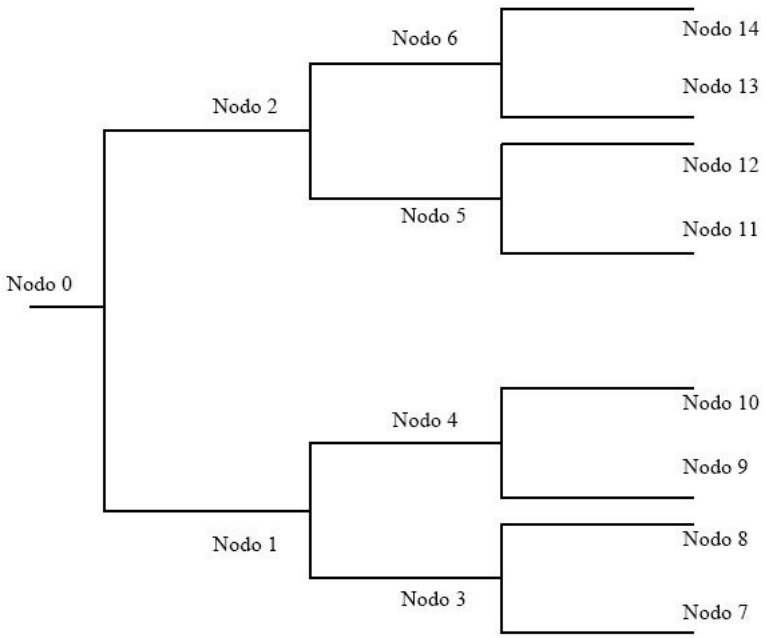


Figura 10: ***Árbol binario (l=2).*** Juárez, C.A. (2004).

d.1).- Búsqueda por el método de Branch and Bound

Cada nodo p se puede probar para determinar si el vecino más próximo a X puede estar en Sp con la aplicación de la Regla 1.

Regla 1: Ningún Xi ∈ Sp puede ser el vecino más próximo de X si:

$$B + r_p < d(X, M_p)$$

B es la distancia al k-ésimo vecino más próximo de X. Inicialmente B

Demostración (Regla 1): Para un Xi ∈ Sp

$d(X,X_i) + d(X_i,M_p) \geq d(X,M_p)$ Desigualdad triangular

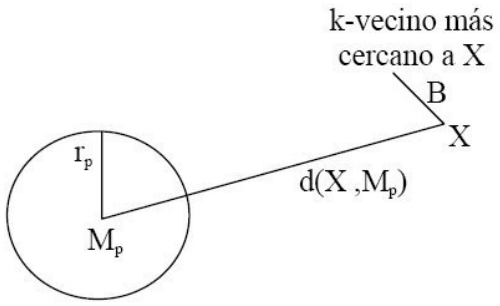


Figura 11: ***K-NN Regla 1.*** Juárez, C.A. (2004).

Puesto que $d(X_i,Mp) \leq r_p$

$$d(X,X_i) \geq d(X,M_p) - r_p$$

Por lo tanto Xi no puede ser el vecino más próximo a X si:

$$d(X,X_i) \geq d(X, M_p) - r_p > B$$

Muchos nodos p y los correspondientes grupos de muestras Sp se pueden eliminar sin calcular explícitamente las distancias a las muestras individuales en Sp. Para un nodo p en el nivel final del árbol, si la regla 1 no se satisface, las distancias a las muestras individuales en Sp desde X se deben calcular. Sin embargo, muchos cálculos de distancias se pueden evitar como sigue.

Regla 2: X_i no puede ser el vecino más próximo a X si:

$$B+d(X_i, M_p) < d(X,M_p)\ X_i \in Sp$$

Demostración (Regla 2): Para un $X_i \in Sp$

$$d(X,X_i)+d(X_i, M_p) \geq d(X,M_p)$$ Desigualdad triangular

$$d(X,X_i) \geq d(X,M_p) - d(X_i,M_p)$$

Por lo tanto X_i no puede ser el vecino más próximo a X si:

$$d(X,M_p) - d(X_i,M_p) > B$$

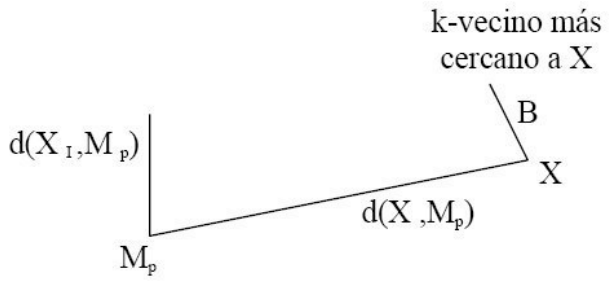


Figura 12: ***K-NN Regla 2.*** Juárez, C.A. (2004).

Se presenta a continuación en p seudo código, el proceso de búsqueda del vecino más próximo empleando el algoritmo de Fukunaga/Narendra.

Función buscar [Fukunaga/Narendra]

Entrada t (árbol)

x (individuo receptor)

Entrada / salida d_{nn} (distancia al vecino más próximo)

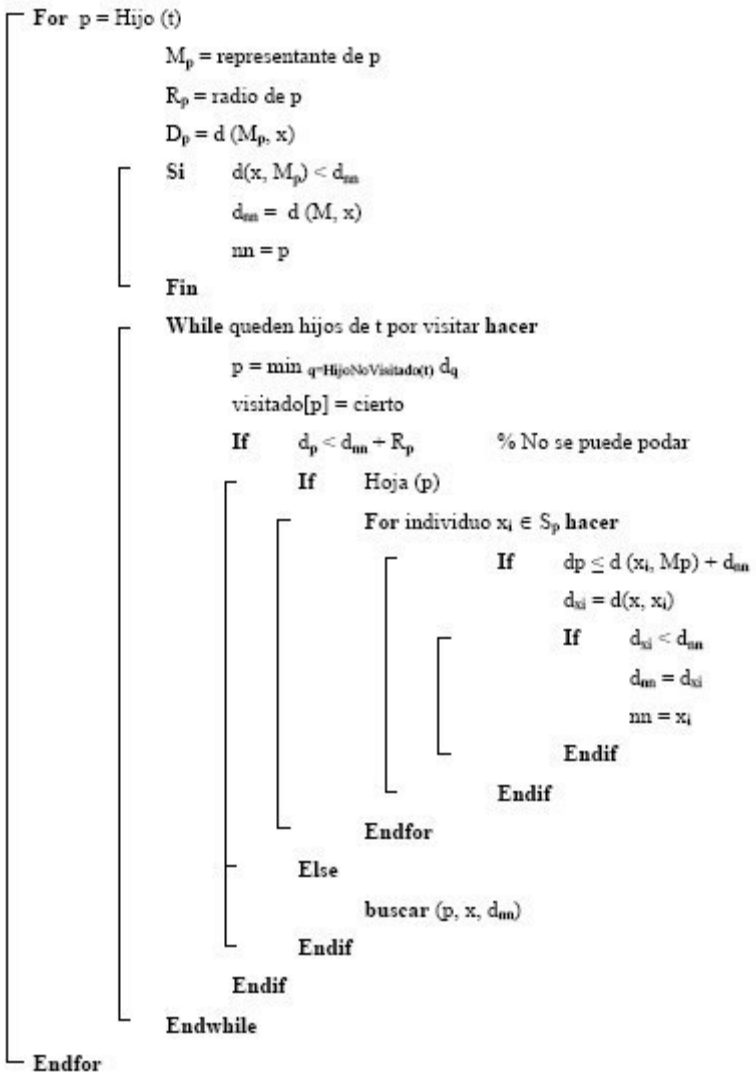


Figura 13: ***Función buscar [Fukunaga/Narendra].*** Juárez, C.A. (2004).

Se pueden encontrar los k vecinos más próximos realizando los siguientes cambios en el algoritmo:

- Cuando se calcula una distancia, se almacena en un vector o lista que contiene los k vecinos más próximos hasta el momento.
- En las condiciones de eliminación o poda, se debe utilizar la distancia al último de los k vecinos más próximos encontrados hasta el momento.

2.2.2 Conglomerados (Clasificación no supervisada)

2.2.2.1 Análisis de Conglomerados Jerárquicos

El análisis de conglomerados (clúster) tiene por objeto agrupar elementos en grupos homogéneos en función de las similitudes o similaridades entre ellos. Normalmente se agrupan las observaciones, pero el análisis de conglomerados puede también aplicarse para agrupar variables. Estos métodos se conocen también con el nombre de métodos de clasificación automática o no supervisada, o de reconocimiento de patrones sin supervisión. El nombre de no supervisados se aplica para distinguirlos del análisis discriminante. El análisis de conglomerados estudia tres tipos de problemas:

Partición de los datos. Disponemos de datos que sospechamos son heterogéneos y se desea dividirlos en un número de grupos prefijado, de manera que:

(1) cada elemento pertenezca a uno y solo uno de los grupos;
(2) todo elemento quede clasificado;
(3) cada grupo sea internamente homogéneo.

*Construcción de jerarquías**.*** Deseamos estructurar los elementos de un conjunto de forma jerárquica por su similitud. Por ejemplo, tenemos una encuesta de atributos de distintas profesiones y queremos ordenarlas por similitud. Una clasificación jerárquica implica que los datos se ordenan en niveles, de manera que los niveles superiores contienen a los inferiores. Este tipo de clasificación es muy frecuente en biología, al clasificar animales, plantas etc. Estrictamente, estos métodos no definen grupos, sino la estructura de asociación en cadena que pueda existir entre los elementos. Sin embargo, como veremos, la jerarquía construida permite obtener también una partición de los datos en grupos.

Clasificación de variables. En problemas con muchas variables es interesante hacer un estudio exploratorio inicial para dividir las variables en grupos. Este estudio puede orientarnos para plantear los modelos formales para reducir la dimensión que estudiaremos más adelante. Las variables pueden clasificarse en grupos o estructurarse en una jerarquía. Los métodos de partición utilizan la matriz de datos, pero los algoritmos jerárquicos utilizan la matriz de distancias o similitudes entre elementos. Para agrupar variables se parte de la matriz de relación entre variables: para variables continuas suele ser la matriz de correlación, y para variables discretas, se construye, como veremos, a partir de la distancia ji-cuadrado.

Problema. Dado un conjunto de n objetos (animales, plantas, minerales...), cada uno de los cuales viene descrito por un conjunto de p características o variables, deducir una división útil en un número de clases “g”. Se han de determinar tanto el número de clases como las propiedades de dichas clases.

Solución. Partición de los “n” objetos (Sujetos) en un conjunto de grupos (g) donde un objeto pertenezca a un grupo sólo y el conjunto de dichos grupos contenga a todos los objetos.

Planteamiento del problema. Sea X una muestra de n individuos sobre los que se miden p variables.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ & \cdots\cdots & & \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix} =$$

	Variables			
Sujetos	X_1	X_2	…	X_p
1	x_{11}	x_{12}	…	x_{1p}
2	x_{21}	x_{22}	…	x_{2p}
…		…….		
n	x_{n1}	x_{n2}	…	x_{np}

X es un conjunto de valores numéricos que se pueden ordenar en una matriz:

x_{11}: Valor que presente el primer sujeto en la primera variable
x_{12}: Valor que presente el primer sujeto en la segunda variable
x_{ij}: Valor que presente el sujeto i - ésimo en la variable j ésima

Cada columna contiene los valores que toman todos los individuos para cada variable que se estudia.

Objetivo. Encontrar una partición de los "n" individuos en "g" grupos de forma que cada individuo pertenezca a un solo grupo y solamente a uno.

2.2.2.2 Conglomerados Jerárquicos mediante vecinos más próximos.

El procedimiento análisis de conglomerados jerárquico permite aglomerar tanto casos como variables y elegir entre una gran variedad de métodos de aglomeración y medidas de distancia. Pero la diferencia fundamental entre ambos procedimientos está en que en el segundo de ellos se procede de forma jerárquica. El análisis de conglomerados jerárquico comienza con el cálculo de la matriz de distancias entre los elementos de la muestra (casos o variables). Esa matriz contiene las distancias existentes entre cada elemento y todos los restantes de la muestra. A continuación se buscan los dos elementos más próximos (es decir, los dos más similares en términos de distancia) y se agrupan en un conglomerado. El conglomerado resultante es indivisible a partir de ese momento: de ahí el nombre de jerárquico asignado al procedimiento. De esta manera, se van agrupando los elementos en conglomerados cada vez más grandes y más heterogéneos hasta llegar al último paso, en el que todos los elementos muéstrales quedan agrupados en un único conglomerado global. En cada paso del proceso pueden agruparse casos individuales, conglomerados previamente formados o un caso individual con un conglomerado previamente formado. El análisis de

conglomerados jerárquico es, por tanto, una técnica aglomerativa, partiendo de los elementos muéstrales individualmente considerados, va creando grupos hasta llegar a la formación de un único grupo o conglomerado constituido por todos los elementos de la muestra.

i.- Distancias Euclídea.

Los métodos jerárquicos parten de una matriz de distancias o similaridades entre los elementos de la muestra y construyen una jerarquía basada en una distancia. Si todas las variables son continuas, la distancia más utilizada es la distancia euclídea entre las variables estandarizadas. No es, en general, recomendable utilizar las distancias de Mahalanobis, ya que la única matriz de covarianzas disponible es la de toda la muestra, que puede mostrar unas correlaciones muy distintas de las que existen entre las variables dentro de los grupos. Si no estandarizamos, la distancia euclídea dependerá sobre todo de las variables con valores más grandes, y el resultado del análisis puede cambiar completamente al modificar su escala de medida. Si estandarizamos, estamos dando a priori un peso semejante a las variables, con independencia de su variabilidad original, lo que puede no ser siempre adecuado. En un espacio de p dimensiones, la fórmula de la distancia euclidiana es:

$$d(x_i, x_j) = \sqrt{\sum_{c=1}^{p} (x_{ic} - x_{jc})^2}$$

Dónde: S_{ic} y X_{jc} son las coordenadas respectivas de los puntos x_i y x_j en la dimensión c.

ii.- Algoritmos Jerárquicos.

Dada una matriz de distancias o de similitudes se desea clasificar los elementos en una jerarquía. Los algoritmos existentes funcionan de manera que los elementos son sucesivamente asignados a los

grupos, pero la asignación es irrevocable, es decir, una vez hecha, no se cuestiona nunca más. Los algoritmos son de dos tipos:

1. De aglomeración. Parten de los elementos individuales y los van agregando en grupos.
2. De división. Parten del conjunto de elementos y lo van dividiendo sucesivamente hasta llegar a los elementos individuales.
3. Los algoritmos de aglomeración requieren menos tiempo de cálculo y son los más utilizados.

iii.- Métodos Aglomerativos

Los algoritmos aglomerativo que se utilizan tienen siempre la misma estructura y sólo se diferencian en la forma de calcular las distancias entre grupos. Su estructura es:

1. Comenzar con tantas clases como elementos, n. Las distancias entre clases son las distancias entre elementos originales.
2. Seleccionar los dos elementos más próximos en la matriz de distancias y formar con ellos una clase.
3. Sustituir los dos elementos utilizados en (2) para definir la clase por un nuevo elemento que represente la clase construida. Las distancias entre este nuevo elemento y los anteriores se calculan con uno de los criterios que comentamos a continuación.
4. Volver a (2) y repetir (2) y (3) hasta que tengamos todos los elementos agrupados en una clase única.

iv.- Criterios para definir distancias entre grupos

Supongamos que tenemos un grupo A con n_a elementos, y un grupo B con n_b elementos, y que ambos se fusionan para crear un grupo (AB) con $n_a + n_b$ elementos. La distancia del nuevo grupo, (AB), a

otro grupo C con n_c elementos, para el caso de la tesis se calcula por la regla siguiente.

v.- Vecino más próximo o encadenamiento simple

La distancia entre los dos nuevos grupos es la menor de las distancias entre grupos antes de la fusión. Es decir:

$$d(C ; AB) = min (d_{CA} , d_{CB})$$

Una forma simple de calcular con un ordenador el mínimo entre las dos distancias es utilizar que

$$mín. (d_{CA}, d_{CB}) = 1/2 (d_{CA} + d_{CB} - |d_{CA} - d_{CB}|)$$

En efecto, si $d_{CB} > d_{CA}$ el término en valor absoluto es $d_{CB} - d_{CA}$ y el resultado de la operación es d_{CA}, la menor de las distancias.

Si $d_{CA} > d_{CB}$ el segundo término es $d_{CA} - d_{CB}$ y se obtiene d_{CB}.

Como este criterio sólo depende del orden de las distancias será invariante ante transformaciones monótonas: obtendremos la misma jerarquía aunque las distancias sean numéricamente distintas. Se ha comprobado que este criterio tiende a producir grupos alargados, que pueden incluir elementos muy distintos en los extremos.

vi.- Árbol jerárquico o dendrograma

El dendrograma, o árbol jerárquico, es una representación gráfica del resultado del proceso de agrupamiento en forma de árbol. Los criterios para definir distancias que hemos presentado tienen la

propiedad de que, si consideramos tres grupos, A, B, C, se verifica que.

$$d(A,C) \leq Max \{d(A,B),D(B,C)\}$$

y una medida de distancia que tiene esta propiedad se denomina ultra métrica. Esta propiedad es más fuerte que la propiedad triangular, ya que una ultra métrica es siempre una distancia.

En efecto si $d^2(A,C)$ es menor o igual que el máximo de $d^2(A,B)$, $d2(B,C)$ forzosamente será menor o igual que la suma $\mathbf{d^2(A,B) + d^2(B,C)}$. El dendrograma es la representación de una ultra métrica, y se construye como sigue:

1. En la parte inferior del gráfico se disponen la n elementos iniciales.

2. Las uniones entre elementos se representan por tres líneas rectas. Dos dirigidas a los elementos que se unen y que son perpendiculares al eje de los elementos y una paralela a este eje que se sitúa al nivel en que se unen.

3. El proceso se repite hasta que todos los elementos están conectados por líneas rectas. Si cortamos el dendrograma a un nivel de distancia dado, obtenemos una clasificación del número de grupos existentes a ese nivel y los elementos que los forman. El dendrograma es útil cuando los puntos tienen claramente una estructura jerárquica, pero puede ser engañoso cuando se aplica ciegamente, ya que dos puntos pueden parecer próximos cuando no lo están, y pueden aparecer alejados cuando están próximos.

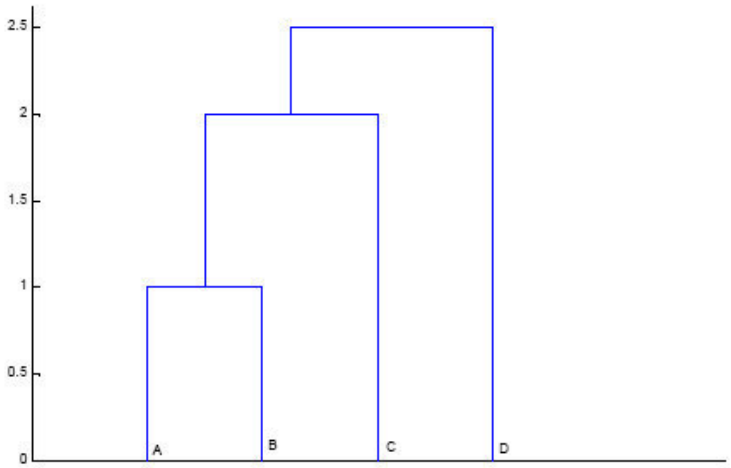


Figura 14: **Árbol jerárquico (Dendrograma) del método vecino más próximo.** Peña, D. (2002)

2.2.3 Algoritmo de k vecinos más próximos K-NN (Clasificación supervisada).

El método de los k-vecinos o k-nn es un método retardado y supervisado (pues su fase de entrenamiento se hace en un tiempo diferente al de la fase de prueba) cuyo argumento principal es la distancia entre instancias. El método básicamente consiste en comparar la nueva instancia a clasificar con los datos k más próximos conocidos, y dependiendo del parecido entre los atributos el nuevo caso se ubicará en la clase que más se acerque al valor de sus propios atributos (cumpliendo así lo planteado por el concepto de heurística de consistencia). La principal dificultad de este método consiste en determinar el valor de k, ya que si toma un valor grande se corre el riesgo de hacer la clasificación de acuerdo a la mayoría (y no al parecido), y si el valor es pequeño puede haber imprecisión en la clasificación a causa de los pocos datos seleccionados como instancias de comparación. Para enfrentar este problema se plantearon diferentes variaciones del método: en cuanto a la forma de determinar el valor de k, por ejemplo 1-nn, que no es otra cosa más

que usar como instancia de comparación al primer vecino más próximo encontrado.

También el valor de k puede hallarse tomando un radio de comparación o mediante el uso de diagramas de Voronoi. Una característica importante e interesante de k-nn es que el método puede cambiar radicalmente sus resultados de clasificación sin modificar su estructura, solamente cambiando la métrica utilizada para hallar la distancia. Por lo tanto, los resultados pueden variar tantas veces como métodos de hallar distancia entre puntos haya. La métrica debe seleccionarse de acuerdo al problema que se desee solucionar. La gran ventaja de poder variar métricas es que para obtener diferentes resultados el algoritmo general del método no cambia, únicamente el procedimiento de medida de distancias.

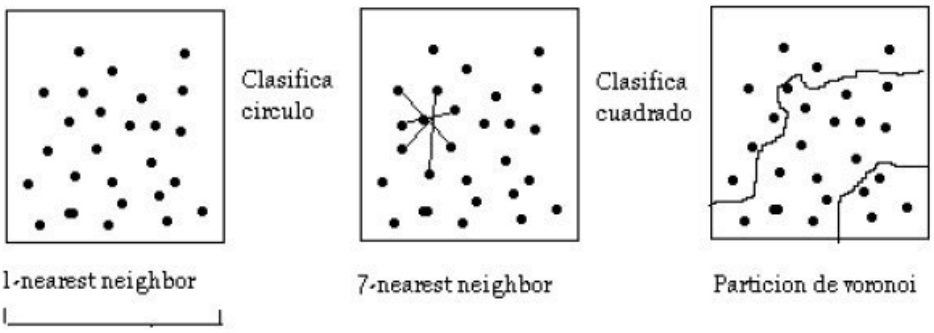


Figura 15: **Diferencia en el valor de k de los vecinos más próximos y partición realizada.** Rodríguez, J., Rojas E. & Franco, R. (2008)

Algoritmo de clasificación por vecindad

- Una técnica de clasificación supervisada
- Precisa de una definición de una métrica que ayude a comparar las distancias entre los objetos.
- Gozan de simplicidad conceptual: la clasificación de un nuevo espacio de representación se calcula en función de las clases, conocidas de antemano, de los puntos más próximos a él. Así las

muestras pertenecientes a una clase se encontrarán próximas en el espacio de representación.

i. *Reglas del vecino más próximo*.

Considerando un espacio de representación, el caso a ser clasificado tomará la clase que esté más cerca dentro del espacio. Dado un conjunto S de n puntos del plano y dado un punto q, hallar el punto de S más próximo a q. Es claro que para resolver el problema basta con hallar las distancias entre q y cada uno de los puntos de S y quedarse con el que dé la menor distancia. Si dispondríamos de unos instrumentos de dibujo o de un sistema de geometría dinámica, podemos plantearles que resuelvan el problema de modo gráfico. Una posibilidad consiste en trazar todas las circunferencias centradas en el punto q que pasan por los puntos de S y seleccionar el punto que determina la más pequeña.

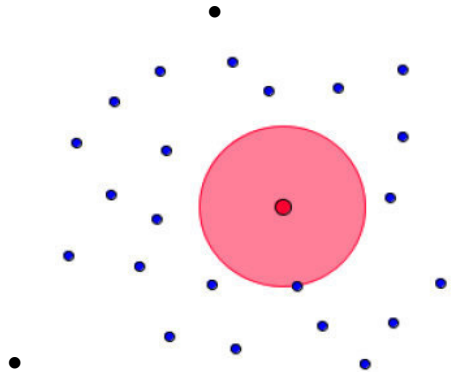


Figura 16: **El vecino más próximo.** Abellanas M. (1993)

La idea de vecindad está asociada a la existencia de una determinada región vacía. En general, dos puntos serán vecinos si no hay ningún punto “entre ellos”, o, dicho de otra forma, si “el espacio entre ellos está vacío”. En los siguientes apartados vamos a usar esta idea para definir de diferentes formas la vecindad entre los puntos de un conjunto. Como el problema ha resultado demasiado sencillo, generalicemos ahora el problema.

ii. Reglas de los K vecinos más próximos.

El nuevo caso a ser clasificado se ubicará en la clase con más votos en el contexto de los K vecinos más cerca del conjunto de entrenamiento. Dado un conjunto S de n puntos del plano, hallar, para cada uno de sus puntos, cuál es el punto de S, distinto de él, más próximo a él.

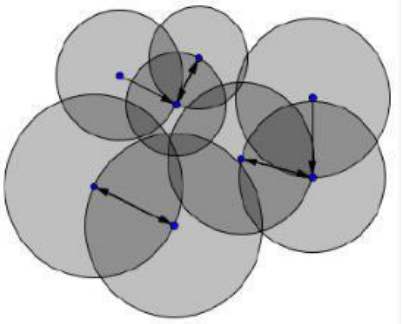


Figura 17: **Todos los vecinos más próximos.** Abellanas M. (1993).

2.2.3.1 K-Vecinos más próximos (K-Nearest Neighbour).

En este tema vamos a estudiar un paradigma clasificatorio conocido como K-Vecinos más próximos (K-Nearest Neighbour). La idea básica sobre la que se fundamenta este paradigma es que un nuevo caso se va a clasificar en la clase más frecuente a la que pertenecen sus K vecinos más próximos. El paradigma se fundamenta por tanto en una idea muy intuitiva, lo que unido a su interesante implementación hace que sea un paradigma clasificatorio muy extendido que consiste, primero en definir una medida de distancia entre puntos, habitualmente la distancia de Euclidiana, segundo calcular las distancias del punto a clasificar, $\mathbf{x_0}$, a todos los puntos de la muestra, tercero seleccionar los **k** puntos muéstrales más próximos al que pretendemos clasificar. Luego calcular la proporción de estos **k** puntos que pertenece a cada una de las poblaciones, finalmente clasificar el punto x_0 en la población con mayor frecuencia de puntos entre los k. Este método se conoce como k−vecinos próximos. En el caso particular de k = 1 el método consiste en asignarle a la población al

que pertenece el elemento más próximo. Un problema clave de este método es claramente la selección de k. Una práctica habitual es tomar $k=\sqrt{n_g}$ donde n_g es un tamaño de grupo promedio. Otra posibilidad es probar con distintos valores de k, aplicárselo a los puntos de la muestra cuya clasificación es conocida y obtener el error de clasificación en función de k. Escoger aquel valor de k que conduzca al menor error observado.

a. Regla K-NN básico.

K-Vecinos más próximos es uno de los métodos de aprendizaje basados en instancias más básicas. La idea en el algoritmo es almacenar el conjunto de entrenamiento, de modo tal que para clasificar una nueva instancia, se busca en los ejemplos almacenados casos similares y se asigna la clase más probable a estos. Un nuevo caso se va a clasificar en la clase más frecuente a la que pertenecen sus k vecinos más próximos. Idea muy simple e intuitiva. Fácil implementación. No hay modelo explícito. Case Based Reasoning (CBR)

Algoritmo K-NN

- COMIENZO
- Entrada: D = {(x1, c1), . . . , (xN, cN)} x = (x1, . . . , xn) nuevo caso a clasificar
- PARA todo objeto ya clasificado (xi, ci) calcular di = d(xi, x)
- Ordenar di(i = 1, . . . ,N) en orden ascendente
- Quedarnos con los K casos D_{KX} ya clasificados más cercanos a x
- Asignar a x la clase más frecuente en D_{KX}
- FIN

b. Variantes del algoritmo K-NN básico

Entre las principales variantes tenemos: K-NN con rechazo, K-NN con distancia media, K-NN con distancia mínima, K-NN con pesado de vecinos, K-NN con pesado de variables.

b.1 K- NN con rechazo

Regla k-NN con rechazo. Se toma en cuenta un nivel fijado con anterioridad que sirve como referencia para que cuando una clase tiene un mayor número de votos que ese nivel, entonces la clase podrá ser asignada. Ese nivel puede tomar un valor entre K/M y K, K es el número de vecinos más próximos y M es el total de clases.

K_NN con rechazo, K=5, U=2: Para clasificar un caso exijo ciertas garantías. Si no lo tengo puedo dejar el caso sin clasificar. Umbral prefijado. Mayoría absoluto.

b.2 K_NN con distancia media.

Regla k-NN por distancia media. Un caso es clasificado en una clase si es que el valor de la distancia media es el menor con respecto al de las otras clases. Ejemplo.

b.3 K_NN con distancia mínima

Clasificador de la distancia mínima. Primero se selecciona un representante para cada clase. Luego la tarea consiste en clasificar al nuevo caso en la clase cuyo vecino es el más próximo al nuevo caso. Seleccionar un caso por clase (ej. El más próximo al baricentro de la clase). Reducción de la dimensión del fichero almacenado de N a m. Ejecutar un 1- NN a dicho fichero reducido. Se toma un caso presentante de cada posible valor, luego se mide distancias y se asigna la clase del que tenga distancia más corta.

b.4 K-NN con pesado de vecinos (casos)

Suponiendo que los casos (x1,&1),… (xk,&k) son los k vecinos más próximos a un caso x a clasificar, en un problema de clasificación con M clases al caso x se le asignara la clasificación más vota de entre

estos vecinos, teniendo en cuenta que cada voto viene ponderado por un peso W_i, i = 1,….k que no tiene que ser el mismo para todos.

Para asociar los pesos a los K vecinos se debe tomar en cuenta que. El voto que aporta un caso dado es inversamente proporcional a la distancia que se encuentra de la instancia a clasificar.

$$w_i = \frac{\bullet 1}{Dist(x, x_i)}$$

Voto fijo según el orden de vecindad. Voto ponderado según las prioridades a priori de las clases a las que pertenece los datos en caso de empate. De este modo, en caso de empate se elige la clase menos probable, ya que es la que menos peso tiene.

b.5 K-NN con pesado de variables

Consideremos un problema de clasificación en el que se ha tenido en cuenta n variables predictores $x_1,x_2,\ldots x_n$, y que el peso atribuido a cada una de esta n variable es $W_1,W_2,\ldots,W_n$. La distancia entre dos casos de este problema, $x=(x_1,x_2,\ldots,x_n)$ e $y=(y_1,y_2,\ldots y_n)$ se calcula.

$$D_w = \sum_{i=1}^{n} W_i (x_i - y_i)^2$$

El peso asociado a cada variable X viene determinado por la medida de la información mutua entre ella y la variable a clasificar C. Si una variable determina exactamente la clase, la información mutua entre ambos es proporcional al logaritmo del número de clases, asumiendo que las instancias de cada una de las clases son igualmente frecuentes. El peso a obtener por la variable será proporcional al valor. Determinar un conjunto de pesos fijos l<n, y asociarse a cada una de las variables, algunos de estos pesos según el resultado de algún tipo de test.

c. Selección de casos

La Técnica de selección de casos se suelen clasificar en; Técnicas encaminadas a eliminar casos erróneamente clasificados del conjunto de entrenamiento, y a la vez, eliminar los posibles solapamientos entre regiones de clases distintas en el espacio de representación (técnica de edición). Técnicas que se centran en seleccionar un subconjunto suficientemente representativo del conjunto de casos inicial (técnicas de condensación o condensada).

c.1. Técnicas de Edición:

Lleva a la organización de casos en clúster pertenecientes a la misma clase.

- **Edición de Wilson:** Utiliza la técnica de validación *Leaving-One-Out,* que consiste en eliminar todos aquellos casos que resulten mal clasificados utilizando la regla K-NN. El principal inconveniente que tiene este algoritmo es el alto costo computacional.

- **Edición repetitiva:** Aprovecha los agrupamientos más o menos compactos que proporciona el método de edición de Wilson. No es una mejora significativa ya que aumenta el costo computacional por ser repetitivo.

- **Edición con re etiquetado:** Se basa en re etiquetar determinados casos en función de la zona de representación en la que se encuentren. En casos reales esta estrategia no representa buenos resultados.

- **Edición con rechazo:** Se utiliza no solo en edición sino en clasificación también. M $\rightarrow$ M +1.

- **Multiedit:** Se aplica repetidamente el proceso de edición por partición pero con: K=1 (1-NN). Los esquemas de edición basados en particiones se usan para conjuntos de muestras amplias.

c.2. Técnicas de Condensado:

Se pretende que la reducción de casos no afecte a la eficiencia del clasificado

- **Condensador de Hart:** Define la consistencia con respecto al conjunto de entrenamientos es consistente con respecto a otro conjunto D, si al utilizar S como conjunto de entrenamiento, es posible clasificar los casos D correctamente.

- **Condensado Reducido:** Se pretende eliminar del conjunto consistente obtenido a partir del condesado de Hart. No asegura la consecución del algoritmo mínima consistente

- **Nearest Neighbour selectivo:** El sub conjunto que se obtiene es el subconjunto más pequeño que contiene al menos una instancia de cada una de las relaciones de este tipo que aparecen en el conjunto de entrenamiento.

- **IB2 E IB3:** IB2 guarda sólo los casos mal clasificados utilizando el algoritmo NN. IB3 utiliza instancias aceptables como casos, se selecciona previamente instancias no ruidosas de la BD y luego se reduce el conjunto resultante. Sólo un subconjunto de estas será utilizado en el proceso de clasificación.

2.2.3.2. Estimación mediante los K vecinos más próximos

Supongamos un espacio de representación bidimensional y una serie de elementos (puntos) de una misma clase representados en él. Dado

un patrón cualquiera X, si consideramos los k puntos (elementos) más próximos a X, éstos estarán localizados en un círculo centrado en X. En la figura 19, resaltamos los 7 vecinos más próximos a tres patrones.

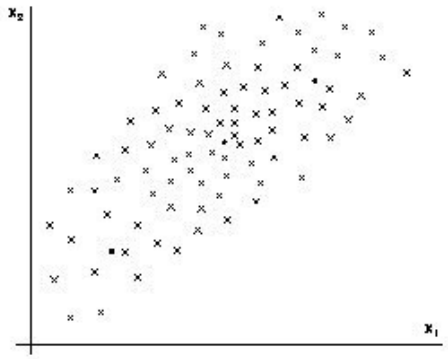


Figura 18: **Patrones en un espacio bidimensional**. Cortijo, F. J. (2001).

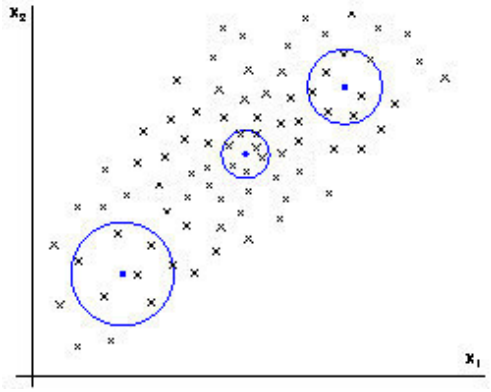


Figura 19: **Los patrones considerados para la estimación de P(X/w) mediante los 7 vecinos más próximos son los encerrados en los círculos.** Cortijo, F. J. (2001).

Parece sensato pensar que el área del círculo que encierra un número fijo de puntos, k, es menor en regiones densamente pobladas que en regiones donde los puntos están más dispersos. Este sencillo planteamiento es la base de la estimación mediante los k vecinos más próximos. En espacios multidimensionales, el círculo se convierte en una híperesfera, y el planteamiento anterior se puede extender

fácilmente ya que el volumen de la híperesfera que encierra a k puntos está relacionado con el valor de la función de densidad de probabilidad en el centro de la híperesfera. Veamos de qué manera.

Si $K_i(X)$ i = 1, 2,..., J es el número de elementos de clase $\boldsymbol{w_i}$ que se encuentran en una vecindad de X, el valor de p^(X/$\boldsymbol{w_i}$) puede calcularse como la proporción de los N_i elementos de la clase $\boldsymbol{w_i}$ que se encuentran en esa vecindad:

$$\hat{P}\ (X|\ \omega_i)\ =\ \frac{K_i(X)}{N_i} \quad \text{.....(1)}$$

Si recordamos la manera en la que se estimaba la densidad de probabilidad empleando núcleos de Parzen, las vecindades son iguales para todos los elementos sobre los que se realiza la estimación y están determinadas por el parámetro. Nuestro objetivo ahora es distinto ya que la vecindad no se establece en base a un parámetro fijado de antemano sino que depende del patrón considerado para la estimación:

Si el patrón se encuentra en una región muy poblada, la vecindad a considerar tendrá un radio menor. Si el patrón se encuentra en una región poco poblada, la vecindad a considerar tendrá un radio mayor.

Con esta idea, la ecuación (1) debe modificarse de manera que se pondere inversamente con el área de la región considerada. Para el caso d-dimensional, el área se convierte en el volumen de una híperesfera centrada en X, V(X), por lo que el nuevo estimador se formula como:

$$\hat{P}\ (X|\ \omega_i)\ =\ \frac{K_i(X)}{N_i V(X)} \quad \text{..... (2)}$$

Para clarificar más acerca de la híperesfera es necesario especificar que es la que está centrada en X y contiene los k vecinos más

próximos a X. En el caso bidimensional (ver figura 1), la esfera se convierte en un círculo y el volumen se convierte en superficie.

Se puede deducir que esta manera de estimar la densidad de probabilidad es diametralmente opuesta a la estimación por núcleos de Parzen ya que los métodos basados en núcleos realizan la estimación con un núcleo de ancho fijo mientras que la estimación por los vecinos más próximos se hace mediante un ``núcleo'' de ancho variable que depende de la densidad de los casos en el espacio de representación.

2.2.3.3. Sobre la elección de K

El problema que se plantea ahora es el de si existe un valor óptimo para k o en su defecto si se pueden dar algunas reglas para fijar este valor. En primer lugar debemos considerar que k juega un papel similar a para los núcleos de Parzen, por lo que resulta evidente que la elección de k está determinado por la densidad de los puntos y debería hacerse en relación a N_i.

Puede demostrarse que el estimador dado en 2 es insesgado y consistente si se verifican las siguientes condiciones sobre k:

$$\lim_{N_i \to \infty} k(N_i) = \infty$$

$$\lim_{N_i \to \infty} k(N_i) / N_i = 0$$

Así, una elección adecuada de k (Ni) es:

$$k(N_i) = cte \sqrt{N_i} \quad \text{..... (3)}$$

2.2.3.4. Métodos de clasificación del vecino más próximo

Para justificar la estimación de la función de densidad por el vecino más próximo, las probabilidades a priori pueden estimarse por la frecuencia relativa global.

$$\hat{\pi}_i = \frac{N_i}{N} \quad \text{..... (4)}$$

y considerando que el estimador de la densidad de probabilidad de p^(X/w_i , un estimador de la probabilidad a posteriori será.

$$\hat{P}(\omega_i|X) = \hat{P}(X|\omega_i)\,\hat{\pi}_i = \frac{K_i(X)}{N_i v(X)}\,\frac{N_i}{N} = \frac{K_i(X)}{v(X)\,N} = \frac{K_i(X)}{k} \quad \text{...(5)}$$

A partir de este resultado se formula la siguiente regla de clasificación:

$$\text{Seleccionar } \omega_c \text{ si } K_c(X) = \max_{i=1\ldots J} \{K_i(X)\}$$

Conocida como regla de clasificación por los k vecinos más próximos o simplemente k-NN (del inglés, k nearest neighbour). Cuando k = 1, la regla anterior se conoce como la conocida como regla de clasificación del vecino más próximo o simplemente 1-NN. Con otras palabras, podemos afirmar que los casos próximos tienden a ser de la misma clase (1-NN) o bien a tener una probabilidad a posteriori similar (k-NN).

Como puede verse, estas reglas proporcionan una estimación directa de la probabilidad a posteriori de cada una de las clases y las reglas de clasificación son sencillas y fácilmente interpretables. A continuación estudiaremos con más detalle las reglas de clasificación por vecindad. Posteriormente abordaremos dos aspectos avanzados sobre estas reglas, considerando en primer lugar la posibilidad de reducir el error de clasificación.

A. Las reglas 1-NN y K-NN

Las reglas de clasificación por vecindad están basadas en la búsqueda en un conjunto de casos de los k casos más próximos al patrón a clasificar.

La búsqueda no se realiza necesariamente en el conjunto completo de casos, T, por lo que denominaremos conjunto de referencia (y lo notaremos por R) al conjunto de casos sobre el que se buscará el(los) vecino(s) más próximo(s). Debemos adelantar que R no tiene porqué ser un subconjunto de T: los métodos adaptativos de aprendizaje proporcionan un conjunto de referencia que no es (usualmente) un subconjunto de T. Sin embargo, los métodos de edición y condensado si proporcionan un subconjunto de T como conjunto de referencia. En cualquier caso, sea cual sea el conjunto de referencia debe especificarse una métrica para poder medir la proximidad. Suele utilizarse por razones computacionales la distancia Euclídea.

A.1.- Regla 1-NN

La regla de clasificación por vecindad más simple es la regla de clasificación del vecino más próximo o simplemente 1-NN. Se basa en la suposición de que la clase del patrón a etiquetar, X, es la de los casos más próximos en R, al que notaremos por X_{NN}. Si | R| = N esta regla puede expresarse como:

$$d(X) = \omega_c \text{ si } \begin{cases} \delta(X, X_{NN}) = \min_{i=1\ldots N}\{\delta(X, X_i)\} \\ (X_{NN}, \omega_c) \in R \end{cases} \quad \ldots.(6)$$

En la figura 6 mostramos cómo se clasificaría el patrón X con la regla 1NN para un problema de clasificación de dos clases. Existen cuatro elementos de clase 1 (representados por cruces) y cinco casos de clase dos (representados por asteriscos). El elemento más próximo es de clase 2, por lo que ésta será la clase asociada a X.

El efecto de esta regla es el de dividir el espacio de representación en N “regiones de influencia", una por cada elemento. Cada una de esas regiones tiene forma poligonal y los bordes corresponden a los puntos situados a igual distancia entre casos. Cada una de estas regiones se conoce como región de Voronoi y la partición poligonal del espacio de representación se conoce como partición de Voronoi. En la figura 7 mostramos la partición de Voronoi asociada a este sencillo problema.

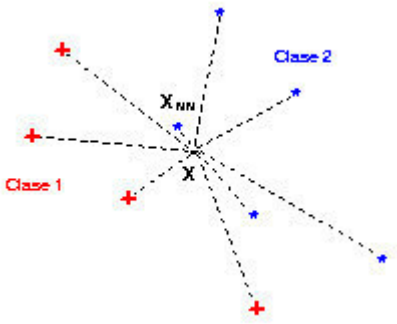


Figura 20: **Clasificación 1-NN.** Cortijo, F. J. (2001).

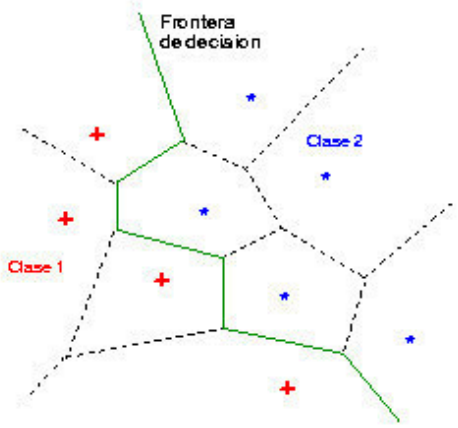


Figura 21: **En trazo continuo, la frontera de decisión; en trazo discontinuo, los bordes de la partición de Voronoi asociada.** Cortijo, F. J. (2001).

Cada región de Voronoi puede considerarse como una región de decisión (restringida a los casos centrales), por lo que la región de decisión de una clase será la unión de todas las regiones de Voronoi de los casos de esa clase. La consecuencia es que las fronteras de decisión serán fronteras lineales a trozos.

A.2.- Regla K-NN

La regla de clasificación por vecindad más general es la regla de clasificación de los k vecinos más próximos o simplemente k-NN. Se basa en la suposición de que los casos más próximos tienen una probabilidad a posteriori similar.

Si $K_i(X)$ es el número de muestras de la clase presentes en los k vecinos más próximos a X, esta regla puede expresarse como:

$$d(X) = \omega_c \quad \text{si} \quad K_c(X) = \max_{i=1\ldots J} \{K_i(X)\} \qquad \ldots\ldots\ldots(7)$$

En la figura 3 mostramos cómo se realizaría la clasificación 3-NN del mismo patrón que se utilizó como ejemplo de clasificación 1-NN en la figura 8. En este caso, K1(X) = 1 y K2(X) = 2 por lo que X se etiquetará como de clase 2.

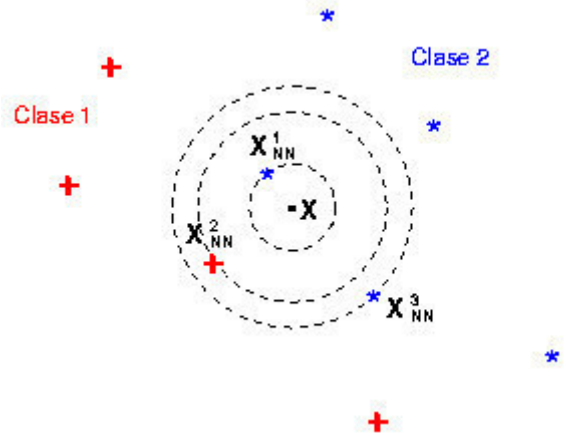


Figura 22: **Clasificación 3-NN.** Cortijo, F. J. (2001)

B. Cotas de error de las reglas 1-NN y K-NN

Partimos de una suposición básica: el conjunto de aprendizaje es grande, virtualmente infinito ($N \to \infty$). En la práctica pocas veces se dispone de un conjunto de esta naturaleza. No obstante, para conjuntos suficientemente grandes pueden asumirse las conclusiones que mostramos en esta sección.

B.1.- Error asociado a la regla 1-NN

Si notamos por E* al error de Bayes y a E1 al error asociado a la regla 1-NN, puede demostrarse que.

$$E^* \leq E_1 \leq E^* \left(2 - \frac{J}{J-1} E^* \right) \quad \text{.....(8)}$$

Donde J es el número de elementos de clase w_i.

Esto es, el error asociado a la regla 1-NN está acotado inferiormente por E* y superiormente por aproximadamente dos veces E*.

B.2.- Error asociado a la regla K-NN

Si notamos por E_k al error asociado a la regla k-NN, puede demostrarse que

$$E^* \leq \ldots \leq E_{2k'+1} \leq E_{2k'} = E_{2k'-1} \leq \ldots \leq E_2 = E_1 \leq 2E^* \quad \text{.....(9)}$$

Donde k/2 ≤ k'. Observar que de esta relación se deduce que $E_1 = E_2$, $E_3 = E_4$, y así sucesivamente, por lo que esta es la razón de que se use habitualmente un número impar para k. Otra consecuencia es que

$$\lim_{k \to \infty} E_k = E^*$$

Esto es, cuanto mayor sea la vecindad considerada, menor será el error asociado a la clasificación k-NN. Esta afirmación debe puntualizarse ya que debe considerarse que el valor de k no puede escogerse tan alto como se desee, sin tener en cuenta el tamaño del conjunto de referencia. Basta pensar en el caso absurdo de considerar k = N (aun siendo N muy grande): siempre se obtendría el mismo resultado.

Consideraciones sobre conjuntos finitos de elementos. La convergencia está teóricamente garantizada en conjuntos con numerosos casos (k/N → 0) y en este caso la decisión sobre la clasificación de X está basada en una pequeña vecindad (en el

sentido volumétrico) de X. Sin embargo, si k/N se incrementa, el volumen de la híperesfera que contiene los k vecinos también, por lo que la convergencia no está asegurada.

B.3.- Tipos de errores y medidas de evaluación.

Primero mostramos la Matriz de confusión para clasificación binaria.

		Valor Observado	
		Clase 1	Clase 2
Predicción	Clase 1	VP	FP
	Clase 2	FN	VN

- Verdaderos positivos (VP): Casos que pertenecen a la clase 1 y el clasificador los definió como 1.
- Falsos positivos (FP): Casos que pertenecen a la clase 1 y el clasificador los definió como 2.
- Falsos negativos (FN): Casos que pertenecen a la clase 2 y el clasificador los definió como 1.
- Verdaderos negativos (VN): Casos que pertenecen a la clase 2 y el clasificador los definió como 2.

$$i)\ Error\ de\ modelo = \frac{N°\ observaciones\ mal\ clasificadas}{N°\ de\ Observaciones\ Totales\ (N)}$$

ii) $$Error\ de\ clasificación = \frac{\boldsymbol{FN+FP}}{\boldsymbol{N}}$$

iii) $$Precisión\ o\ Exactitud = \frac{\boldsymbol{FN+FP}}{\boldsymbol{N}} = 1 - \text{Error de clasificación}$$

C. Extensiones: clase de rechazo

Las reglas de clasificación por vecindad pueden ampliarse para considerar la clase de rechazo. La extensión directa de la regla k-NN dada por la expresión 4 sería la llamada **regla (k, t)-NN**:

$$d(X) = \begin{cases} \omega_c & \text{si } K_c(X) = \max_{i=1...J}\{K_i(X)\} \geq t \\ \omega_0 & \text{en otro caso} \end{cases} \quad \text{.....(10)}$$

En este caso se trata de que la clase más representada en los k vecinos más próximos tenga un número de representantes mayor que un umbral t. En otras palabras, se trata de obtener una mayoría cualificada.

Si se desea especificar un umbral específico para cada clase, la regla anterior se conoce como la **regla (k, t_c)-NN**:

$$d(X) = \begin{cases} \omega_c & \text{si } K_c(X) = \max_{i=1...J}\{K_i(X)\} \geq t_c \\ \omega_0 & \text{en otro caso} \end{cases} \quad \text{..... (11)}$$

y permite un control más estricto sobre el ``grado de confianza'' para aceptar una clasificación en determinadas clases críticas.

Sobre la regla 1-NN no hay más opción que establecer el umbral en base a criterios de distancia. **La regla 1-NN (t)** se formula como sigue:

$$d(X) = \begin{cases} \omega_c & \text{si } \begin{cases} \delta(X, X_{NN}) = \min_{i=1...N}\{\delta(X, X_i)\} \leq t \\ (X_{NN}, \omega_c) \in R \end{cases} \\ \omega_0 & \text{en otro caso} \end{cases} \quad \text{... (12)}$$

La selección del valor umbral t se realiza después de examinar la distribución de las distancias de cada patrón a clasificar al vecino más próximo en R. Parece razonable que el valor umbral sea un valor próximo al tercer cuartil de esta distribución. De nuestra experiencia podemos indicar que el valor exacto depende del problema, esto es, de los datos disponibles y del objetivo buscado, por lo que el valor exacto se selecciona ``ad-hoc'' después de examinar varios candidatos.

2.2.4 Validación cruzada o cross-validation.

Es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y reserva. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Se utiliza cuando el objetivo principal es la predicción y se quiere estimar la precisión de un modelo.

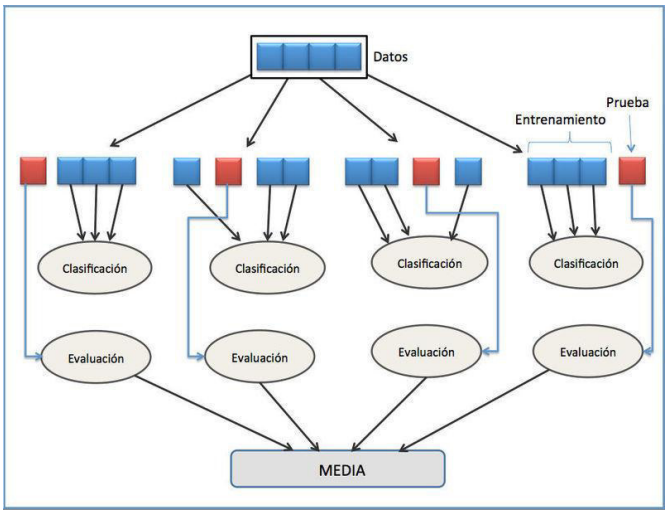


Figura 23: **Esquema k-fold cross validation, con k=4 y un solo clasificador.** Jean-Philippe L, (2003).

El objetivo de la validación cruzada consiste en estimar el nivel de ajuste de un modelo a un cierto conjunto de datos de reserva independientes de las utilizadas para entrenar el modelo. Estas medidas obtenidas pueden ser utilizadas para estimar cualquier medida cuantitativa de ajuste apropiada para los datos y el modelo. La validación cruzada sólo produce resultados significativos si el conjunto de validación y reserva se han extraído de la misma población.

2.2.4.1 Contexto

La validación cruzada proviene de la mejora del método de retención (holdout method). Este consiste en dividir en dos conjuntos

complementarios los datos de muestra, de forma que la función de aproximación sólo se ajusta con el conjunto de datos de entrenamiento y a partir de aquí calcula los valores de salida para el conjunto de datos de reserva. La evaluación puede depender en gran medida de cómo es la división entre datos de entrenamiento y de reserva, y por lo tanto puede ser significativamente diferente en función de cómo se realice esta división. Debido a estas carencias aparece el concepto de validación cruzada.

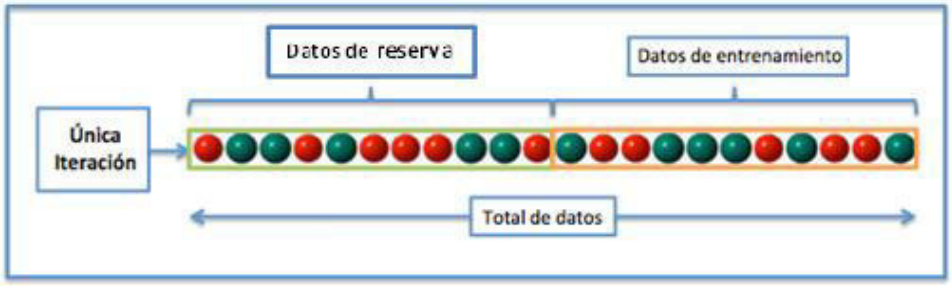


Figura 24: **Método de retención.** Wikipedia (2016)

2.4.4.2 Objetivo de la validación cruzada.

El proceso optimiza los parámetros del modelo para que éste se ajuste a los datos de entrenamiento. Si tomamos una muestra independiente como dato de reserva (validación), del mismo grupo que los datos de entrenamiento, regularmente el modelo no se ajustará a los datos de reserva igual de bien que a los datos de entrenamiento. Esto se denomina sobreajuste y acostumbra a pasar cuando el tamaño de los datos de entrenamiento es pequeño o cuando el número de parámetros del modelo es grande. La validación cruzada es una manera de predecir el ajuste de un modelo a un hipotético conjunto de datos de reserva cuando no disponemos del conjunto explícito de datos de reserva.

2.2.4.3 Tipos de validaciones cruzadas.

Existes varios tipos de validaciones cruzadas entre ellos:

A.- Validación cruzada de K iteraciones

K iteraciones (*K-fold cross-validation)* los datos de muestra se dividen en K subconjuntos. Uno de los subconjuntos se utiliza como datos de reserva y el resto (K-1) como datos de entrenamiento. El proceso de validación cruzada es repetido durante k iteraciones, con cada uno de los posibles subconjuntos de datos de reserva. Finalmente se realiza la media aritmética de los resultados de cada iteración para obtener un único resultado. Este método es muy preciso puesto que evaluamos a partir de K combinaciones de datos de entrenamiento y de reserva. En la práctica, el número de iteraciones depende de la medida del conjunto de datos. Lo común es utilizar 10 iteraciones.

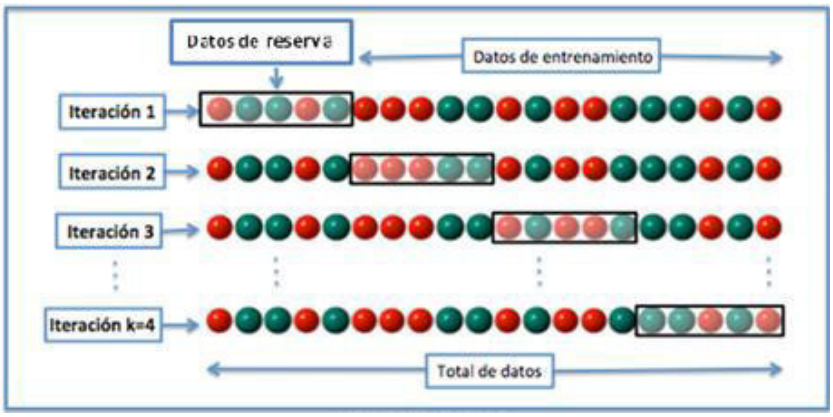


Figura 25: **Validación cruzada de K=4 iteraciones.** Wikipedia (2016)

B.- Validación cruzada aleatoria.

Consiste dividir aleatoriamente el conjunto de datos de entrenamiento y el conjunto de datos de reserva. Para cada división la función de aproximación se ajusta a partir de los datos de entrenamiento y calcula los valores de salida para el conjunto de datos de reserva. El resultado final se corresponde a la media aritmética de los valores obtenidos para las diferentes divisiones. La ventaja de este método es que la división de datos entrenamiento-reserva no depende del número de iteraciones. Pero, en cambio, con este método hay algunas muestras que quedan sin evaluar y otras que se evalúan más de una vez.

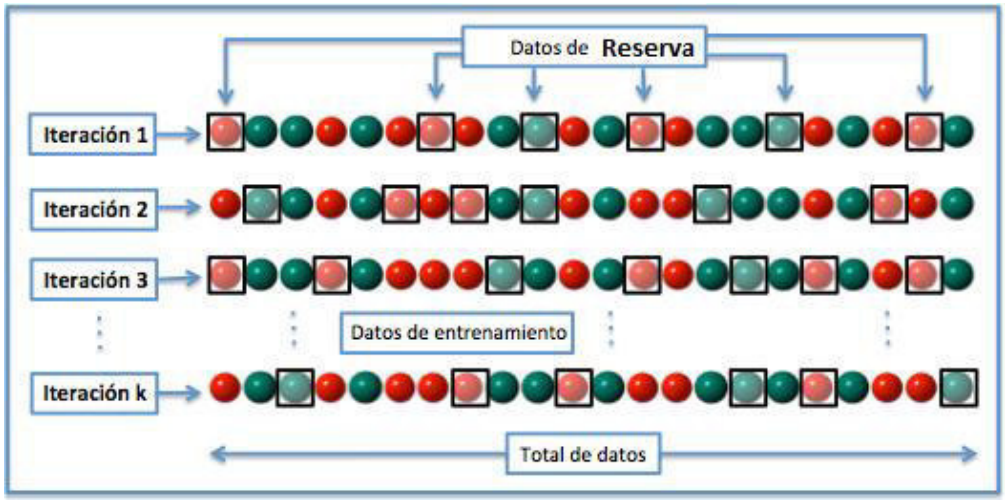


Figura 26: **Validación cruzada aleatoria con K iteraciones.** Wikipedia (2016)

C.- Validación cruzada dejando uno fuera.

Validación cruzada dejando uno fuera *Leave-one-out cross-validation)* implica separar los datos de forma que para cada iteración tengamos una sola muestra para los datos de reserva y todo el resto para entrenamiento. La evaluación viene dada por el error, y en este tipo de validación cruzada el error es muy bajo, pero en cambio, a nivel computacional es muy costoso, puesto que se tienen que realizar un elevado número de iteraciones, tantas como N muestras tengamos y para cada datos tanto de entrenamiento como de reserva.

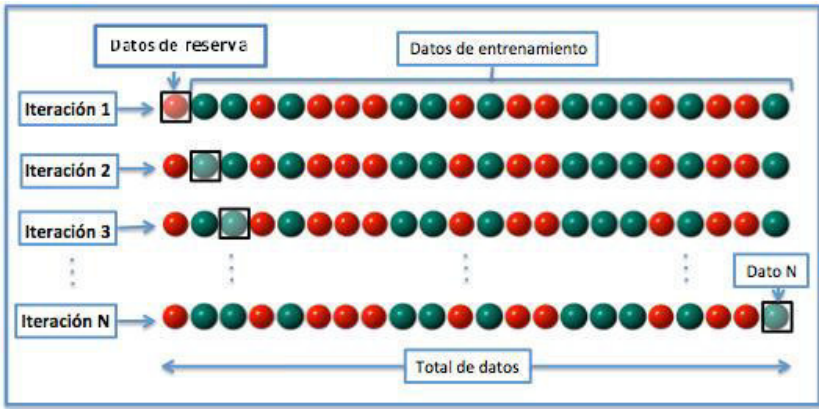


Figura 27: **Validación cruzada dejando uno fuera.** Wikipedia (2016)

2.2.4.4 Cálculo del error promedio

Depende del tipo de validación cruzada.

A.- Error de la validación cruzada de K iteraciones

En cada una de las k iteraciones se realiza un cálculo de error. El resultado final lo obtenemos a partir de realizar la media aritmética de los K valores de errores obtenidos, según la fórmula:

$$E = \frac{1}{K}\sum_{i=1}^{K} E_i.$$

B.- Error de la validación cruzada aleatoria

Cogemos muestras al azar durante k iteraciones, se realiza un cálculo de error para cada iteración. El resultado final también lo obtenemos a partir de realizar la media aritmética de los K valores de errores.

$$E = \frac{1}{K}\sum_{i=1}^{K} E_i.$$

C.- Error de la validación cruzada dejando uno fuera

Se realizan tantas iteraciones como muestras (N) tenga el conjunto de datos. De forma que para cada una de las N iteraciones se realiza un cálculo de error. El resultado final lo obtenemos realizando la media aritmética de los N valores de errores obtenidos.

$$E = \frac{1}{N}\sum_{i=1}^{N} E_i.$$

2.2.5 Pruebas no paramétricas para k muestras independientes

Para analizar datos provenientes de diseños con una variable independiente categórica (más de dos grupos) y una variable dependiente cuantitativa en la cual interesa comparar las muestras. La prueba H de Kruskal-Wallis y la prueba de la mediana.

2.2.5.1 Kruskal – Wallis.

Si J muestras son aleatoria e independientemente extraídas de J poblaciones para averiguar si las J poblaciones son idénticas o alguna de ellas presenta promedios mayores que otra. Las ventajas: no necesita establecer supuestos sobre las poblaciones originales tan

exigentes como los del estadístico F (normalidad, homocedasticidad). Consideremos J muestras aleatorias e independientes de tamaños $n_1,n_2,...,n_J$ extraídas de la misma población o de J poblaciones idénticas. Llamemos n al conjunto total de observaciones: $n=n_1+n_2+...+n_J$. Asignemos rangos desde 1 hasta n a ese conjunto de n observaciones como si se tratara de una sola muestra (si existen empates se asigna el promedio de los rangos empatados). Llamemos R_{ij} a los rangos asignados a las observaciones i de la muestra j. Llamemos Rj a la suma de los rangos asignados a las n_j observaciones de la muestra j. Tendremos:

$$R_j = \sum_{i}^{n_j} R_{ij} \qquad \text{y} \qquad \bar{R}_j = \frac{R_j}{n_j}$$

Si la hipótesis nula de que las J poblaciones son idénticas es verdadera, los Rj de las distintas muestras serán parecidos.

$$H = \frac{12}{n(n+1)} \sum_{j=1}^{J} \frac{R_j^2}{n_j} - 3(n+1)$$

Bajo la hipótesis nula de que los J promedios poblacionales son iguales, el estadístico H se distribuye según el modelo de probabilidad Chi-cuadrado, con J-1 grados de libertad.

$$\mathbf{H_0} : \mu_1 = \mu_2 = ... = \mu_I$$
$$\mathbf{H_1} : \mu_i \neq \mu_j \text{ para algún par } (i,j)$$

En el caso de que existan empates. Donde k se refiere al número de rangos distintos en los que existen empates y ti al número de valores empatados en cada rango.

$$H' = \frac{H}{1 - \sum_{i=1}^{k} (t_i^3 - t_i)/(n^3 - n)}$$

2.2.5.2 Prueba de la Mediana

La prueba de la mediana es similar a la prueba chi-cuadrado. La única diferencia entre ambas es que ahora, en lugar de utilizar dos variables

categóricas, una de ellas es cuantitativa y se dicotomiza utilizando la mediana (de ahí el nombre de la prueba). Tenemos, por tanto, una variable categórica que define J muestras de tamaño n_j ($n=\Sigma n_j$) y una variable al menos ordinal.

El objetivo de la prueba de la mediana es contrastar la hipótesis de que las J muestras proceden de poblaciones con la misma mediana.

$$H_0 : M_1 = M_2 = ... = M_I$$
$$H_1 : M_i \neq M_j \text{ para algún par}(i, j)$$

Para ello, se comienza ordenando todas las observaciones y calculando la mediana total (la mediana de las n observaciones):

$$Mdn = (X_{[n/2]} + X_{[n/2+1]}) / 2 \qquad \text{si } n \text{ es par}$$
$$Mdn = X_{[(n+1)/2]} \qquad \text{si } n \text{ es impar}$$

Donde $X_{[n]}$ se refiere al valor más grande y $X_{[1]}$ al más pequeño.
Por último, se aplica el estadístico chi-cuadrado

$$X^2 = \sum_i \sum_j \frac{(n_{ij} - m_{ij})^2}{m_{ij}}$$

2.2.6 Algoritmo de predicción mediante K vecinos más próximos

Supongamos que tenemos una nueva Corte Superior de Justicia o es más una determinada Corte Superior por diversos motivos cambio sus valores de sus variables en estudio, el problema consiste en predecir este caso a que grupo pertenecerá. Para solucionar este problema planteamos el siguiente algoritmo en RStudio.

Algoritmo 1

- Se elige un número de vecinos próximos (k).
- Se elige una métrica, es decir, una función para calcular la distancia (Euclidiana).
- Para cada ejemplo Corte Superior de Justicia (CSJ):
  - Se calcula la distancia al resto de los ejemplos.
  - Se seleccionan los k vecinos más próximos.
  - La clase de CSJ es la más representada entre estos k.
  - Resolución de empates. Si coincide el número de vecinos de dos o más clases, se escoge la clase con mayor probabilidad a priori. Si las probabilidades a priori coinciden, se escoge una de las clases en disputa al azar.

Algoritmo 2

COMIENZO
Entrada: $D = \{(x_1;c_1), .. (x_N,c_N)\}$
$x = (x_1,..,x_n)$ ***nuevo caso a clasificar***
PARA todo objeto ya clasificado $(x_i; c_i)$
Calcular $d_i = d(x_i; x)$
Ordenar $d_i(i = 1,….,N)$ en orden ascendente
Quedarnos con los K casos D^K_x ya clasificados más próximos a x
Asignar a x la clase más frecuente en D^K_x
FIN

Modelo que predice nuevos casos (Función en R)

```
Tesis<-read.delim("clipboard")   [Pegar]
attach(Tesis)
library(class)
X<-Tesis[2:7]
C<-t(Tesis[8])
C<-Tesis[,8]
Y<-matrix(c(0.30,0.24,0.22,0.25,0.20,0.16),nrow=1)
knn(X, Y, C, k = 3, prob=TRUE)
```

Primer Resultado:

```
Y<-matrix(c(0.30,0.24,0.22,0.25,0.20,0.16),nrow=1)
knn(X, Y, C, k = 3, prob=TRUE)
```

Solución:

Grupo:

[1] 3

2.3 Hipótesis y variables

2.3.1 Hipótesis General y específicas

Hipótesis General

- Los modelos construidos mediante el método de los k−vecinos más próximos son precisos para clasificar las 31 Cortes Superiores de Justicia del País y realizar predicciones futuras.

Hipótesis especificas

- Los modelos construidos para los predictores (variables) basado en el método de los k vecinos más próximos es eficaz para clasificar y predecir las Cortes Superiores de Justicia.

- El modelo de k vecinos más próximos se ejecute con precisión para los datos de las variables, cuando se tiene muestras pequeñas de entrenamiento y reserva.

- Los modelos descriptivos permiten identificar y evaluar a las 31 Cortes Superiores de Justicia, respecto de los predictores (variables) en forma a priori.

- El modelo de agrupamiento jerárquico mediante encadenamiento simple (vecinos más próximos) permite asociar a las Cortes Superiores de Justicia del País en tres conglomerados.

2.3.2 Identificación de variables

A continuación se definen cada una de las variables que intervienen en el estudio.

- ***Pendientes.-*** Es la representación de un inventario físico en el período inicial en cada una de las Corte Superior de Justicia que sumados representan los pendientes del Poder Judicial. Es la cantidad de expedientes principales que se encuentran en trámite sin resolución final que concluya el proceso en la instancia y en ejecución sin resolución ejecutada. Asimismo, en trámite son los expedientes que se encuentran en Fiscalía, Consulta a instancia superior o en casos similares, y los expedientes principales reingresados después de haber sido impugnada su resolución final. Además, se debe considerar los expedientes reservados de la especialidad penal.

- ***Ingresos.-*** Es la cantidad de expedientes principales equivalentes a al incremento de la carga procesal en el mes, en cada una de las Corte Superior de Justicia que sumados representan los ingreso del Poder Judicial. Existen 3 tipos de ingresos durante el mes: ingreso de expedientes a trámite, ingreso de expedientes a Ejecución y Otros ingresos. Cada uno se sub divide en las formas que permitan evaluar cualitativamente el aumento de la carga procesal.

- ***Resueltos.-*** Es la cantidad de expedientes principales que implican la disminución de la carga procesal en el mes, en cada una de las Corte Superior de Justicia que sumados representan los resueltos del Poder Judicial. Puede ser representativa de la etapa de trámite o de la etapa de ejecución. En el primer caso, es la cantidad de expedientes principales resueltos (fallados) en la instancia mediante una sentencia, auto final o informe final (exclusivo en los juzgados penales con procesos ordinarios). También contiene la resolución revisoría o de segunda instancia de las apelaciones a las resoluciones emitidas

en una instancia inferior y que elevan al expediente principal. El segundo caso se trata de la ejecución de la resolución final. En este caso considera a cada órgano jurisdiccional como un representante del Poder Judicial que cumple su rol ante la sociedad en forma definitiva, archivando un expediente principal definitivamente.

- ***Población.-*** Esta representado por el grupo de personas que viven en un área o espacio geográfico llamada Corte Superior de Justicia, es decir, es la cantidad de habitantes existentes en cada una de las Corte Superior de Justicia que sumados representan la población del País en el año 2013.

- ***Órganos jurisdiccionales***.- Representa la cantidad de dependencias jurisdiccionales existentes en cada Corte Superior de Justicia, que sumadas constituyen las dependencias jurisdiccionales del Poder Judicial.

- ***Personal.-*** Representa la cantidad de personas (personal administrativo y jurisdiccional) que laboran en cada una de las Corte Superior de Justicia, que sumandos constituyen los trabajadores del Poder Judicial del País.

CARGA PROCESAL.- La carga procesal es la cantidad total de procesos judiciales principales que obran en cada órgano jurisdiccional y que se encuentran en Trámite o en Ejecución. Se obtiene del resultado de la siguiente suma:

CARGA PROCESAL = PENDIENTES + INGRESOS

III. METODOLOGÍA

3.1. Tipo y Diseño de la Investigación

El tipo de investigación, según el nivel de medición y análisis de la información es descriptivo, correlacional e inferencial, debido a que se realizan análisis de los datos mediante gráficos, indicadores de relación y pruebas estadísticos, con el objeto de identificar las relaciones existen entre las 31 Cortes Superiores de Justica del Perú y sus respectivas variables (Pendientes, Ingresados, Resueltos, Personal, Dependencias, Población). De otro lado se debe precisar que los datos se recogieron sobre la base de las hipótesis y teorías planteadas, con la intención de exponer la información de manera cuidadosa y analizar minuciosamente los resultados obtenidos, con el propósito de extraer generalizaciones significativas que contribuyan al conocimiento dentro y fuera del Poder Judicial. Todos estos resultados y análisis se fundamentan en la información obtenida de una base de datos del Poder Judicial del Perú, llamado Sistema Integrado Judicial – Formulario Estadístico Electrónico. El propósito principal del estudio es que el modelo de clasificación y predicción encontrado sea preciso cuando es aplicado en la práctica en el Poder Judicial.

3.2. Población de estudio

La población de estudio está representada por todas las Cortes Superiores de Justicia del Perú (Poder Judicial del Perú) que funcionaron desde el 01 de enero del año 2013 al 31 de enero del año 2013, que suman un total de treintaiuno (31) Cortes Superiores de Justicia.

De otro lado se debe precisar que la cantidad de Cortes Superiores de Justicia se modifican de acuerdo al año que se pretenda estudiar.

3.3. Unidad de análisis

Para el presente estudio la unidad de análisis (casos de estudio) está representada por cada Corte Superior de Justicia del Perú.

3.4. Tipo y selección de muestra

Para el presente estudio se tomó la información del año 2013 respecto de las todas Cortes Superiores de Justicia que funcionaron desde el 01 de enero del año 2013 al 31 de enero del año 2013. En consecuencia no se realizó ningún tipo de muestreo, es decir, el estudio se realizó sobre la población existen en el respectivo año.

3.5. Técnicas de Recolección de Datos

Los datos se ha extraído directamente de la base de datos del Poder Judicial, llamado "Sistema Integrado Judicial (SIJ) – Formulario Estadístico Electrónico (FEE)", que fue proporcionado por la Sub Gerencia de Estadística de la Gerencia de Planificación del Poder Judicial del Perú. Los datos se recopilan mediante los formularios de información Estadística conocidos con los nombres de "SIJ-FEE-1" y "SIJ-FEE-2".

En el formulario "SIJ-FEE-1" se registra la variable "Pendientes", que representa la cantidad de Proceso Judicial acumulados de períodos anteriores, y los "Ingresos" representan los Procesos Judiciales ingresados durante todo el año 2013. En el formulario "SIJ-FEE-2" de Producción Jurisdiccional, se registra la variable "Resueltos" que representa los procesos resueltos por las dependencias jurisdiccionales, así como la variable "Dependencia" que representa la cantidad

dependencias jurisdiccionales y la variable “Personal” que representa la cantidad de trabajadores.

Respecto a la variable “Población”, las cifras para cada una de las Corte Superior de Justicia es una estimación que tiene como base las cifras del Instituto Nacional de Estadística e Informática (INEI). Se debe precisar que existe diferencia geográfica entre cada una de las Cortes Superiores de Justicia y Regiones geográficas del Perú.

La información del Poder Judicial es registrada en forma permanente por los órganos jurisdiccionales (dependencias) mediante los siguientes instrumentos de recolección.

Sistema Integrado Judicial (SIJ) es un software informático que genera la información de los datos registrados mediante el proceso de asociación de “hitos estadísticos” (resoluciones judiciales que señala un estado del proceso) que luego se migra al Formulario Estadístico Electrónico (FEE). Para el caso de órganos jurisdiccionales que no cuenten con el Sistema Integrado Judicial deberá registrar la información mediante el aplicativo web denominado Formulario Estadístico Electrónico.

Finalmente, se debe precisar que la información estadística que se utilizó en el estudio es la contenida en la base de datos del Poder Judicial, debido a que se cuenta con acceso continuo a la información estadística jurisdiccional.

3.6. Procedimiento de análisis de datos

El presente estudio desarrolla los siguientes procedimientos. Primero con el propósito de identificar y evaluar a las 31 Cortes Superiores de Justicia se realiza un análisis descriptivo de los datos con el objeto de encontrar

una apreciación *a priori* del número posible de conglomerados (grupos) mediante el gráfico de estrellas y verificar la variación de los datos mediante análisis de datos atípicos (Box Plot) que localiza los valores atípicos (outlier).

En segundo lugar, mediante conglomerados jerárquicos (clasificación no supervisada) se encuentra un modelo de agrupamiento *a posteriori***,** el cual da como resultado la formación de tres grupos (pequeño, mediano y grande), para ello se calculó la de matriz de distancia euclidiana (valora las distancias), luego apoyado en el método de encadenamiento simple o vecino más próximo se encuentra el historial de conglomeración (mide el grado jerarquía) y se construye el dendrograma o árbol de clasificación, que permite encontrar el mejor modelo a posteriori de agrupamiento,

En tercer lugar, mediante el método de los k-vecinos más próximos (clasificación supervisada) se encuentra el modelo de clasificación y predicción más preciso, mediante la estimación *a priori* del valor k (depende del tamaño del grupo), luego se define la partición de entrenamiento y reserva (para validar el modelo), los pliegues para la validación cruzada (influye en la estimación y predicción para muestras pequeñas), luego se desarrolla el algoritmo (k-NN) que presenta el espacio de predictores (muestra el modelo encontrado), gráficos de homólogas (sitúa al caso focal y sus vecinos), importancia del predictor (indica la importancia de las variables), tabla de vecinos y distancias (valora las distancias), mapas de cuadrantes (analiza el promedio), resumen de errores (precisión del modelo), tabla de clasificación (presión del modelo), el gráfico de registros de errores de selección del valor aposteriori de k (mejor valor de k) y clasificación 3-NN (grupos, probabilidades y dispersión). Además para ratificar la precisión del modelo de 3-vecinos más próximos utilizaremos las pruebas estadísticas no paramétricas de Kruskal-Wallis y la Prueba de la Mediana. Finalmente se presenta un modelo predictivo para clasificar futuras Corte Superiores de Justicia en el Poder Judicial.

IV. RESULTADOS Y DISCUSIÓN

4.1 Análisis exploratorio de datos.

El análisis de datos se realiza con el propósito de identificar y evaluar a las 31 Cortes Superiores de Justicia, respecto de la evaluación *a priori* del número de conglomerados (grupos) mediante el gráfico de estrellas y localizar los valores atípicos (outlier) mediante el análisis de datos atípicos (Box Plot).

4.1.1 Evaluación a priori de los grupos (conglomerados)

El gráfico de estrellas permite evaluar en forma *a priori* los posibles grupos a formarse como se observa en la Figura 28 que muestra a las 31 Cortes Superior de Justicia en el plano, mediante una estrella construida sobre un círculo con las seis variables en estudio (pendientes, ingresos, resueltos, personal, dependencias y población), igualmente espaciados, cuyas magnitudes nacen desde el centro del círculo.

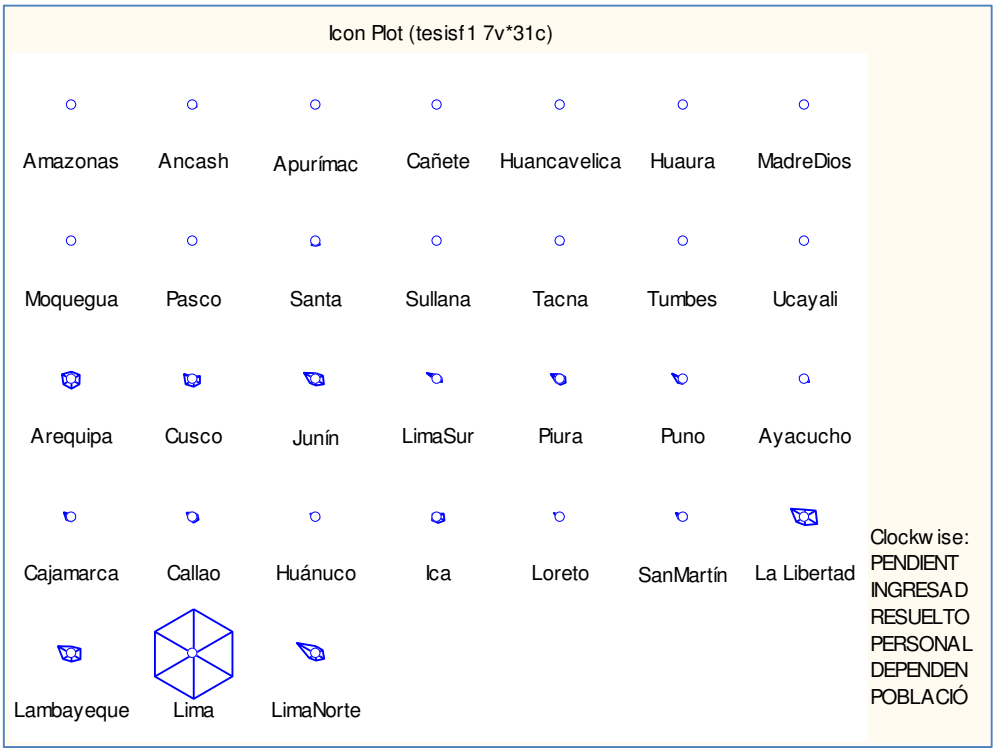


Figura 28. **Grupos a priori de las Cortes Superiores de Justicia.** Statistica 7.

De esta forma la Figura 28, proporciona una idea a priori del número posibles de grupos (conglomerados) que se pueden formar con las 31 Cortes Superiores de Justicia, es así que se puede proponer una representación ***a priori*** de los grupos, apoyados en la magnitud de las seis variables. Que el primer grupo de Cortes Superiores está constituido por: Amazonas, Ancash, Apurímac, Ayacucho, Cañete, Huancavelica, Huánuco, Huaura, Madre Dios, Moquegua, Pasco, Santa, Sullana, Tacna, Tumbes, Ucayali. El segundo grupo: Cajamarca, Callao, Lima Sur, Loreto, Puno, San Martín. El tercer grupo: Arequipa, Cusco, Ica, Junín, La Libertad, Lambayeque, Lima Norte, Piura.

Mientras que, si observamos la gráfica de la Corte Superior Justicia de Lima, podemos distinguir claramente que la representación de las magnitudes de sus variables en el gráfico de estrellas, es la más amplia, respecto de las demás Cortes Superiores, esto sugiere la conformación de un nuevo grupo (grupo cuatro) con un único elemento (Corte Superior de Justicia).

4.1.2 Análisis de Datos Atípicos (Box Plot).

Para identificar los valores atípicos (outlier) se utiliza el gráfico de datos atípicos (Box Plot) que se observa en la Figura N° 29 que muestra los valores extremos en cada una de las variables en estudio.

Igualmente, se puede advertir que existe un dato atípico (valor extremo) representado por la Corte Superior de Justicia de Lima. Como se revela este valor extremo ocurre en las seis variables (pendientes, ingresos, resueltos, personal, dependencias y población) del presente trabajo de investigación.

De otro lado el gráfico box plot de la Figura 29, evidencia que la Corte Superior de Justicia de Lima simboliza un valor extremo, es decir, es una

Corte Superior de Justicia, que es numéricamente distante del resto de las Cortes Superiores, en cada una de las seis variables en estudio.

Asimismo, este valor extremo (atípico) puede tener un efecto desproporcionado en los resultados estadísticos, lo que puede conducir a interpretaciones engañosas en la presente tesis. Estos valores extremos en las variables justifican que la Corte de Lima debe ser tratada de forma especial y única.

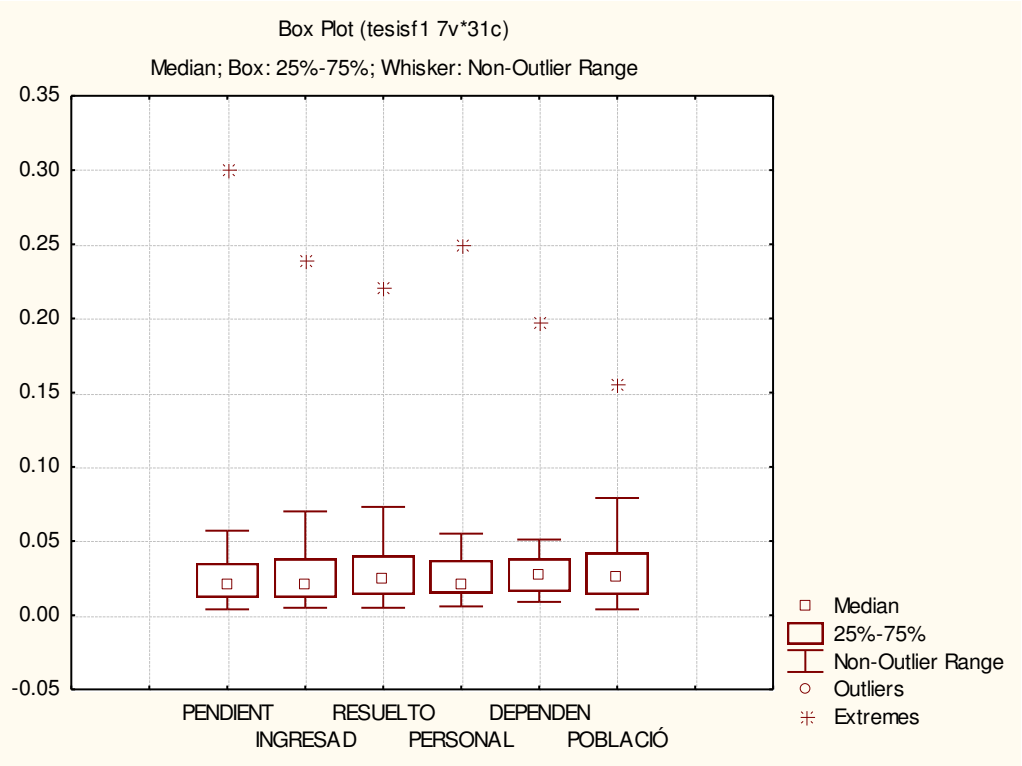


Figura 29. ***Valores extremos en cada variable.*** Statistica 7.

Finalmente apoyados en el análisis del gráficos de estrellas y datos atípicos (Box Plot) mostrados en las Figura N° 28 y Figura N° 29, se observa que las características de la información que ostenta la Corte Superior de Justicia de Lima son muy diferentes respecto a la magnitud de cada una de las variables comparado con las demás Cortes Superiores de Justicia del País.

En consecuencia se hace muy necesario para el presente trabajo de investigación que la formación de conglomerados (grupos) o clasificación no supervisada se realice con las restantes 30 Cortes Superiores de Justicia del Perú.

4.2 Análisis de conglomerados a posteriori (clasificación no supervisada)

La Figura N° 30 muestra el modelo *a posteriori* de los tres grupos (Pequeño, Mediano y Grande) de las Cortes Superiores de Justicia formados de acuerdo al método de conglomerados jerárquicos (Clasificación no supervisada), es decir, para lograr constituir los grupos; primero se calculó la de matriz de distancia euclidiana (Ver anexo 2) que permite valorar las distancias que existen entre cada una de las Cortes Superiores de Justicia, luego apoyado en el método de encadenamiento simple o vecino más próximo se encuentra el historial de conglomeración (Ver Anexo 3) que mide el grado de jerarquía de cada una de las Cortes Superiores, este procedimiento permite construir el dendrograma o árbol de clasificación (Ver anexo 4). Toda la secuencia de este procedimiento se realizó con el propósito de encontrar el modelo de agrupamiento, es decir formar los tres grupos.

De otro lado utilizando el árbol de clasificación ó dendrograma se encuentra el modelo *a posteriori* de agrupamiento, que permite formar los grupos del trabajo de investigación; para lograr el número de conglomerados (grupos) una técnica muy utilizada, consiste en trazar una línea vertical en el gráfico (dendograma) que genera intercepto con los índices (distancias) ordenados de acuerdo a una jerarquía de cada Corte Superior de Justicia, en consecuencia si nos centramos en cada intercepto y sus respectivas ramificaciones son los que representan un determinado grupo. Esto se puede observar en la Figura 30.

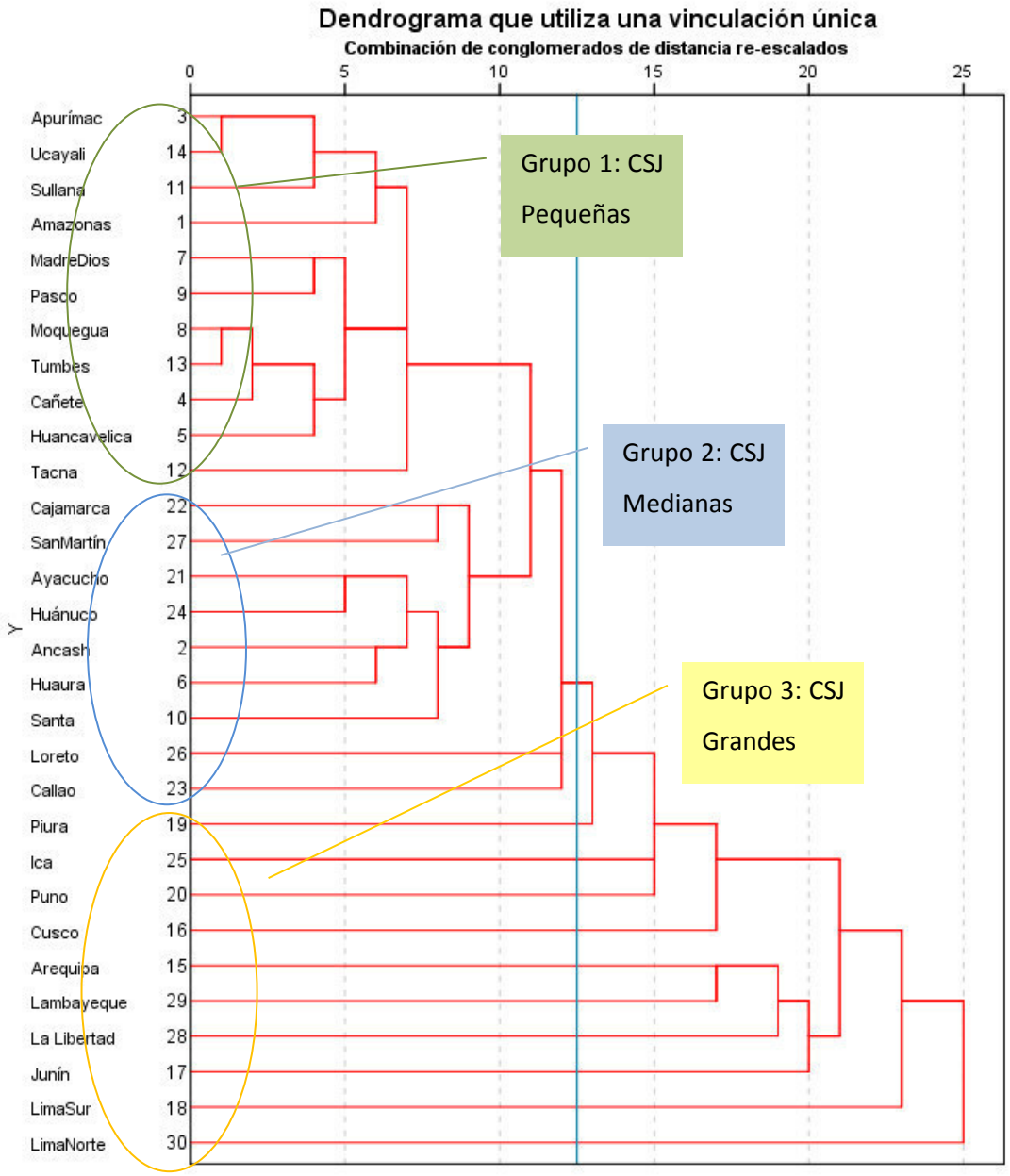


Figura 30: **Árbol jerárquico de Cortes Superiores de Justicia.** SPSS 20.

El Cuadro N° 1, muestra la descripción del modelo *a posterior* encontrado mediante conglomerados jerárquicos utilizando el vecino más próximo. Aquí se detalla los conglomerados formados: Grupo Pequeño; de tamaño n_1=11 Cortes Superiores de Justicia, mientras que el Grupo Mediano n_2= 9 Cortes Superiores de Justicia finalmente el

Grupo Grande n_3=10 Cortes Superiores de Justicia. El tamaño de cada grupo es muy importante porque permite una estimación *a priori* del valor de "k". Este modelo se construye utilizando las seis variables (pendientes, ingresos, resueltos, personal, dependencias y población).

Cuadro 1: **Grupos formados mediante vecinos más próximos**

Cortes Superiores de Justicia				
Grupos	**CSJ Pequeña**	1		Apurímac
		2		Ucayali
		3		Sullana
		4		Amazonas
		5		Madre Dios
		6		Pasco
		7		Moquegua
		8		Tumbes
		9		Cañete
		10		Huancavelica
		11		Tacna
		Total	**n_1**	**11**
	CSJ Mediana	1		Cajamarca
		2		San Martín
		3		Ayacucho
		4		Huánuco
		5		Ancash
		6		Huaura
		7		Santa
		8		Loreto
		9		Callao
		Total	**n_2**	**9**
	CSJ Grande	1		Piura
		2		Ica
		3		Puno
		4		Cusco
		5		Arequipa
		6		Lambayeque
		7		La Libertad
		8		Junín
		9		Lima Sur
		10		Lima Norte
		Total	**n_3**	**10**
	Total	**n**		**30**

La gráfica de dispersión que se observa en la Figura N° 31 presenta el esquema de la variable resueltos versus la variable ingresados en un espacio bidimensional de la agrupación realizada mediante conglomerados jerárquicos, utilizando el vecino más próximo (Cuadro N° 1), donde los puntos representan las Cortes Superiores de Justicia en sus respectivos grupos como son: pequeño, mediano y grande. De otro lado como se observa existe solapamientos entre los grupos, es decir, algunas Cortes grandes están contenidos en el grupo de CSJ mediana, así una corte pequeña el grupo de Corte mediana.

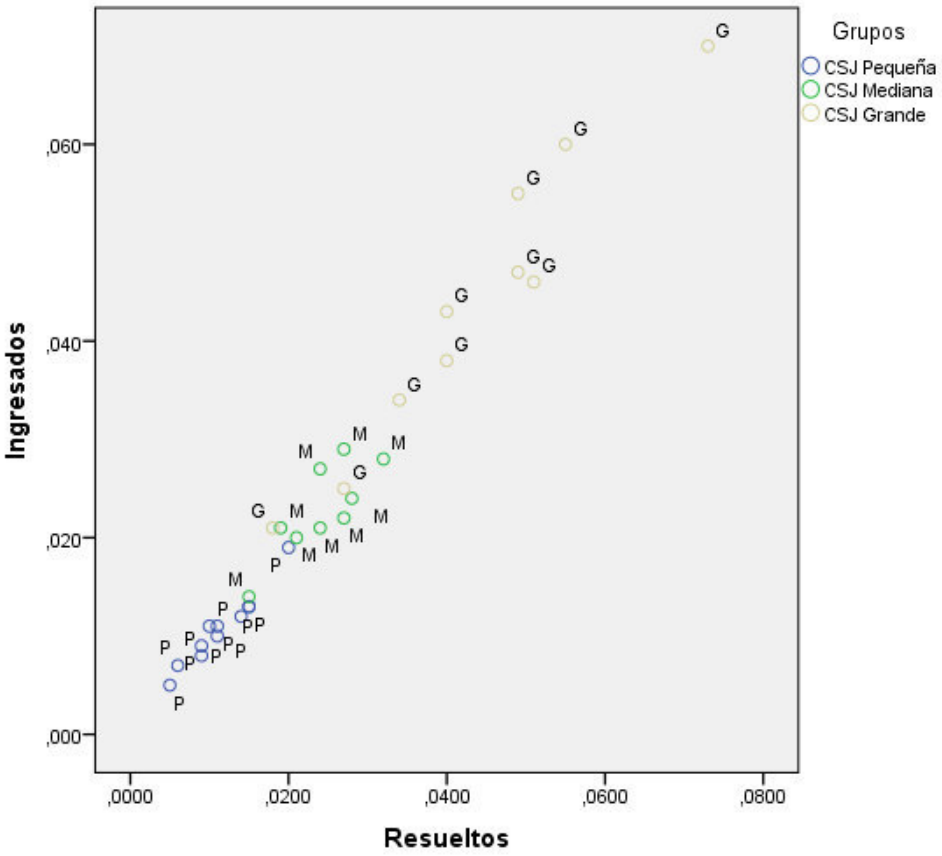


Figura 31: **Gráfica de Grupos; resuelto versus ingresado.** SPSS 20.

Con el fin de mejorar este agrupamiento a continuación utilizaremos la clasificación supervisada mediante método de los k vecinos más próximo. Se debe precisar que para el desarrollo de la tesis se utiliza 6 variables, en consecuencia este gráfico solo es referente.

4.3 Validación del modelo mediante método de los k vecinos más próximos K – NN (Clasificación supervisada).

El algoritmo para validar el modelo es el K vecino más próximo *(K-Nearest Neighbour)* y la secuencia es la siguiente

4.3.1 Estimación a priori del valor de K (K-NN)

La estimación *a priori* del valor de "k" depende fundamentalmente del tamaño del grupo, es decir, del número de Cortes Superiores asignados a cada uno de los conglomerados (grupos) formados en el modelo de agrupamiento *a posteriori* encontrado (Cuadro 1). A continuación se realiza la estimación *a priori* de los tres valores de k.

Conglomerado pequeño:

n_1=11, entonces $k \cong \sqrt{11} = 3.3$

Conglomerado mediano:

n_2=9, entonces $k \cong \sqrt{9} = 3$

Conglomerado grande:

n3=10, entonces $k \cong \sqrt{10} = 3.2$

Cuadro 2: **Valor a priori de k estimado en cada grupo.**

Conglomerado	n_i	k
Pequeño	11	3.3
Mediano	9	3
Grande	10	3.2

Se observa en el Cuadro 2, que los valores *a priori* de k estimado en los tres conglomerados tiende al valor de tres (3). Pero de otro lado se sabe por experiencia anteriores se ha obtenido buenos resultados para valores de "k" igual 3, 4, 5.

En consecuencia para el presente estudio, se tomarán como posibles valores de “k” en el experimentos para encontrar el modelo basado en k vecinos más próximo, los valores de K=3, K=4 y K=5. El valor tres (k=3) por el resultado de la estimación *a priori* y el valor cinco por experiencias anteriores.

4.3.2 Descripción de las Particiones para entrenamiento y reserva

Con el propósito de validar el modelo a encontrar mediante el método de k vecinos más próximos, se realiza la partición de los datos. El Cuadro N° 3, muestra las particiones del conjunto de datos correspondientes a las muestras de entrenamiento y reserva. La muestra de entrenamiento comprende 24 (80%) Cortes Superiores de Justicia utilizados para entrenar y obtener el modelo de vecino más próximo; y la muestra de reserva con 6 (20%) Cortes Superiores que se utiliza para evaluar el modelo final encontrado. De otro lado se debe precisar que la partición de entrenamiento y reserva se realiza mediante asignación aleatoria de las Cortes Superiores.

Cuadro N° 3: **Particiones para entrenamiento y reserva.**

Resumen del procesamiento de los casos

		N	Porcentaje
Muestra	Entrenamiento	24	80,0%
	Reserva	6	20,0%
Válidos		30	100,0%
Excluidos		0	
Total		30	

4.3.3 Pliegues de validación cruzada aleatoria

El objetivo principal del modelo encontrado es la predicción y estimar con precisión cuando se llevará a cabo en la práctica. Además debido a que el tamaño de los datos de entrenamiento y reserva es pequeño, es

necesario utilizar validación cruzada de pliegue. En el presente trabajo de investigación la validación cruzada divide a los datos (muestra) en 3 (tres) pliegues (sub muestras). De otro lado se debe precisar que en el método se asignó las Cortes Superiores de Justicia a los pliegues aleatoriamente. Además las iteraciones que se presentan para conseguir el modelo final se describe de la siguiente manera: el primer modelo se basa en todos los casos excepto los correspondientes al primer pliegue de la muestra; el segundo modelo se basa en todos los casos excepto los del segundo pliegue de la muestra; el tercer modelo se basa en todos los casos excepto los del tercer pliegue de la muestra. El número de tres pliegues (sub muestras) establecido al presente trabajo de investigación, es debido a que se está trabajando con población y grupos pequeños. Esto es 30 Cortes Superiores de Justicia y los grupos son de tamaño: Pequeño n_1=11 mientras que el Mediano n_2= 9 y el Grande n_3=10 Cortes Superiores de Justicia, un pliegue mayor a 3 ocasiona menor número de iteraciones.

4.3.4 Resultados del algoritmo K-NN

A continuación se presentan los resultados del modelo encontrado para el valor de k=3 y se utiliza el valor de K=5 para realizar la comparación de los dos modelos encontrados. De otro lado se debe precisar que antes de llegar al modelo descrito, se realizaron múltiples simulaciones para distintos valores de K.

4.3.4.1. Modelo construido (Espacio de predictores) para k=3 vecinos más próximos

La Figura N° 32, presenta el modelo construido para los predictores (variables), para tres vecinos más próximo (k=3), con sus respectivas particiones de entrenamiento y reserva, el caso focal que está representado por la Corte Superior Justicia de Junín con sus

respectivos vecinos más próximos, y se exhibe los grupos (conglomerados) CSJ pequeña, CSJ mediana y CSJ grande.

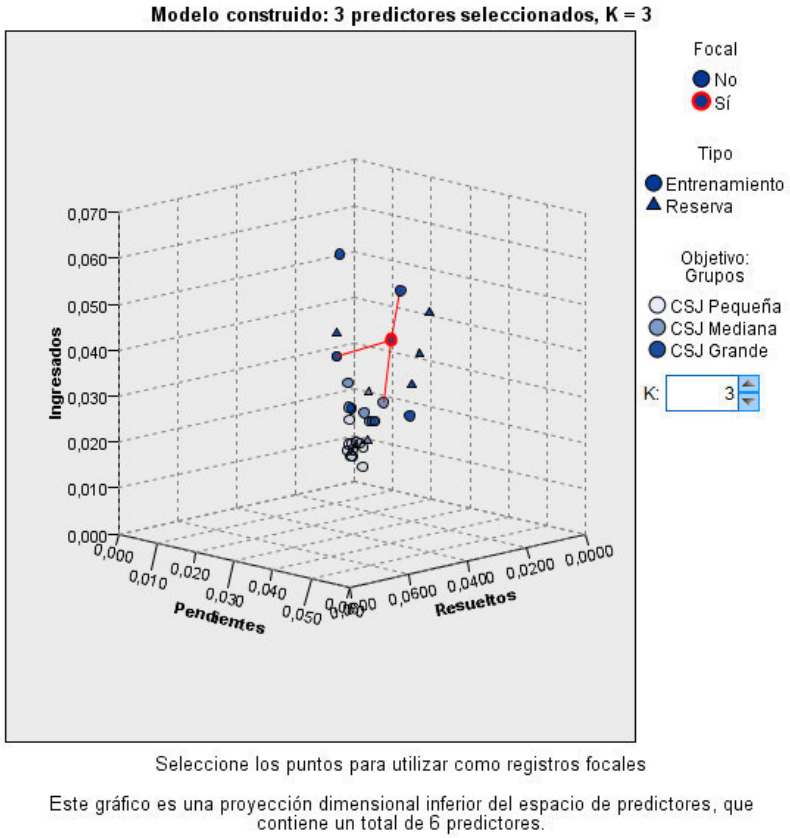


Figura 32: **Modelo construido.** SPSS 20.

En la imagen se observa que la CSJ Junín (focal) se clasifica como CSJ Grande, para el valor de k=3, esto se evidencia debido a que sus dos vecinos más próximos (Ica, Lambayeque) pertenece a la CSJ Grande y un vecino más próximos (Callao) pertenece a la CSJ Mediana, apoyado en el algoritmo utilizado para el modelo de clasificación, la CSJ de Junín (caso focal) se clasifica en el grupo que tiene mayor probabilidad, dado que la probabilidad de grupo grande es mayor que la probabilidad del grupo mediano. Por tanto Junín pertenece al grupo de tamaño grande.

4.3.4.2.- Gráfico de homólogos para k=3 vecinos más próximos

Figura N° 33, este gráfico muestra el caso focal (CSJ Junín) y sus tres (k=3) vecinos más próximos Ica, Lambayeque y Callao, La barra "Grupos" muestra que Junín (focal), Ica y Lambayeque pertenecen al grupo grande (3), mientras que Callao pertenece al grupo mediano (2), las barras de los predictores (variables) muestra el valor que tienen el caso focal (CSJ Junín) y sus tres vecinos más próximos (Ica, Lambayeque y Callao) en cada una de las variables como son Pendientes, Ingresados, Resueltos, Personal, Dependencias. Por ejemplo en la barra "Pendientes" que detalla la ubicación de cada una de las Cortes respecto a su valor en la variable pendiente, se observa que Junín está más cerca de Lambayeque, pero más lejos que Callao e Ica.

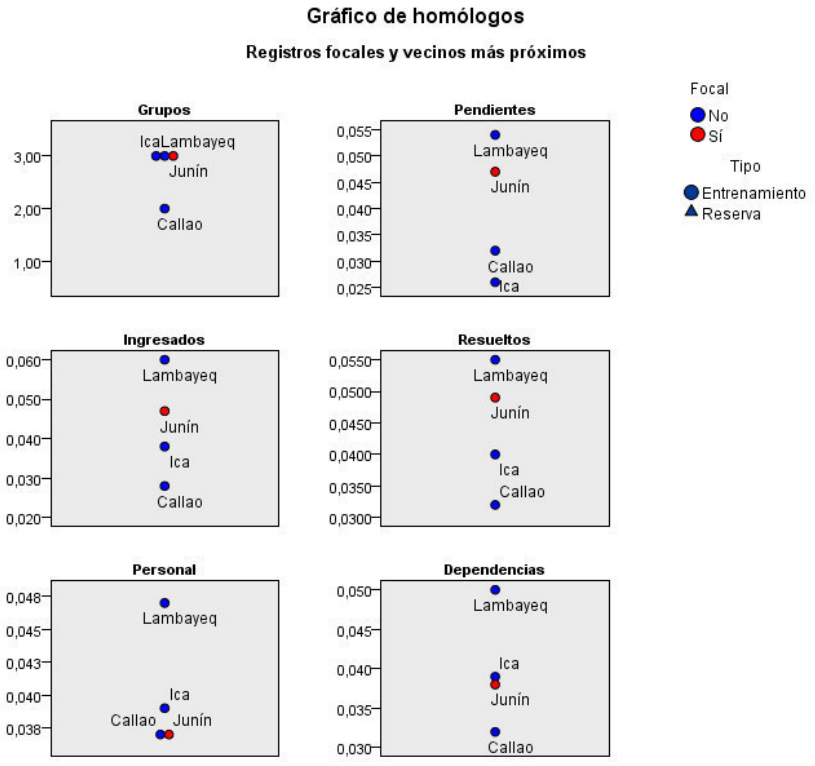


Figura 33: **Gráfico de homólogos k=3.** SPSS 20.

4.3.4.3.- Importancia del predictor para k=3 vecinos más próximos

La Figura N° 34, muestra el gráfico de importancia relativa de las variables en estudio, se observa que la variable pendiente es la más importante en la estimación del modelo ya que su índice de importancia es cercano al 20%, seguida por las demás variables Población, Dependencia, Personal, Resueltos e Ingresados, todos con un índice de importancia cercano al 16%. Además la suma de los indicadores de importancia de todas las variables en estudio debe sumar Uno (100%). De otro es necesario manifestar que este gráfico sólo está relacionado con la importancia de cada una de las variables para realizar un pronóstico. Es decir, es independiente de si éste pronóstico es preciso o no.

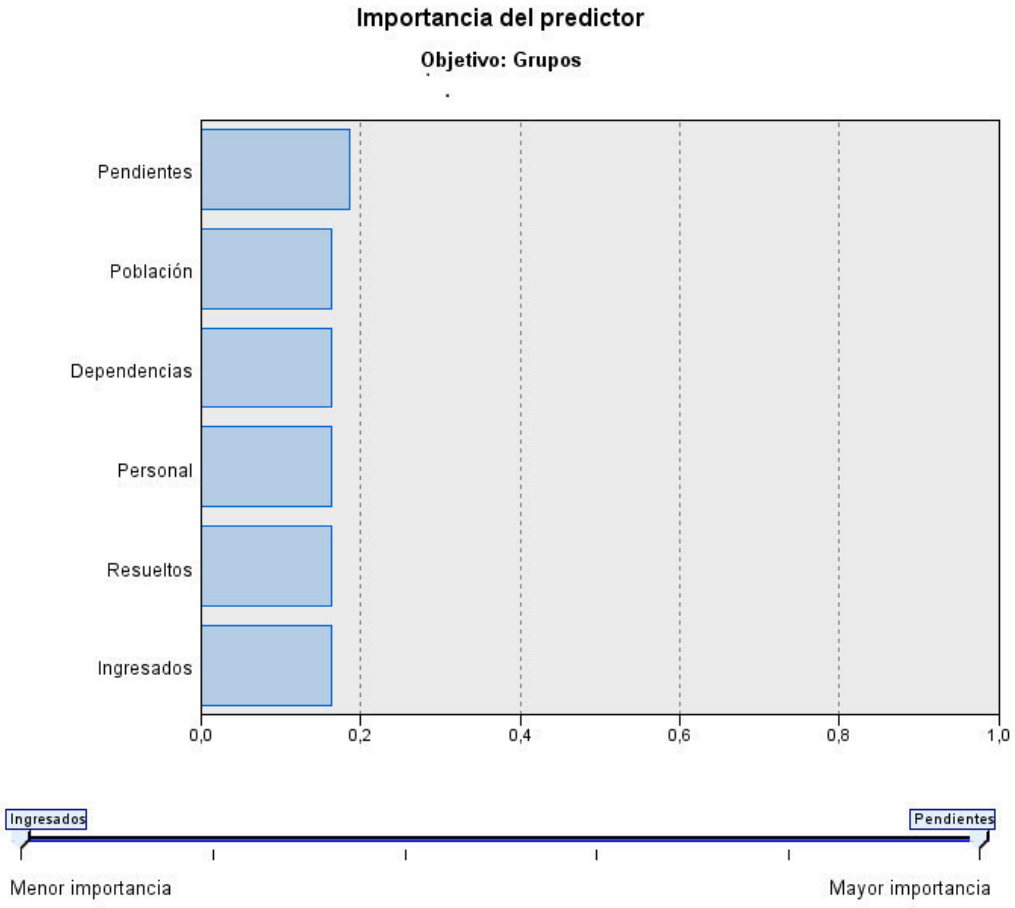


Figura 34: **Gráfico de Importancia del predictor.** SPSS 20.

4.3.4.4.- Tabla de vecinos y distancias para k = 3

El Cuadro 4, muestra la CSJ Junín (focal) y sus tres vecinos más próximos y las distancias más próximas (distancia euclidiana estandarizada) desde CSJ Junín a cada uno de sus vecinos más próximos Lambayeque, Callao e Ica. Respecto de la CSJ de Junín se observa que el primer vecino más próximo es Lambayeque cuya distancia tiene un valor de 0.373, el segundo vecino más próximo es Callao con una distancia de 0.504 y el tercer vecino más próximo es Ica con una distancia de 0.540.

Cuadro 4: **Vecinos más próximos y distancias para k=3.** SPSS 20.

Vecinos más próximos k y distancias

Mostrado para los registros focales iniciales

Registros focal	Vecinos más próximos			Distancias más próximas		
	1	2	3	1	2	3
Junín	Lambayeq	Callao	Ica	0,373	0,504	0,540

Asimismo, es necesario revelar que se puede renovar el caso focal, es decir, considerar otra Corte Superior como caso focal y realizar un análisis similar a lo señalado anteriormente cuando Junín era caso focal.

4.3.4.5.- Mapa de cuadrantes para k=3 vecinos más próximos

La Figura 35, muestra los valores de objetivo por predictores para los registros focales y vecinos más próximos iniciales, donde se observa que la CSJ de Junín (focal) y sus tres vecinos más próximos representados en un diagrama de dispersión (o gráfico de puntos) con el destino (grupos) en el eje vertical (y) y las variables en el eje horizontal (x).

Del mismo modo, si nos centramos en el mapa de cuadrantes respecto de la variable pendiente se evidencia que la CSJ de Ica y CSJ de Callao se encuentran situados debajo de la media (líneas punteadas) mientras que CSJ de Junín y CSJ de Lambayeque se encuentran estacionados en la parte superior de la media, lo mismo sucede con las variables ingresados y resueltos, mientras que para las variables personal y dependencias se muestra que la CSJ de Ica, CSJ Callao y CSJ Junín (Focal) se encuentran colocados debajo de la media (líneas punteadas) mientras que CSJ de Lambayeque se sitúa en parte superior de la media.

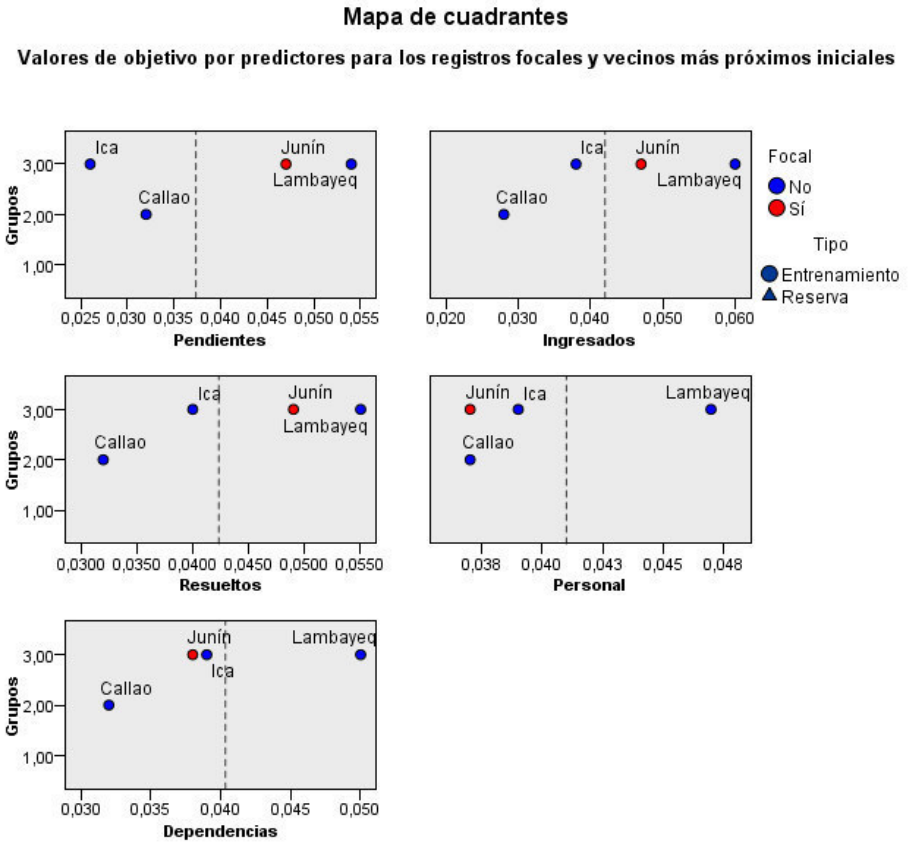


Figura 35: **Mapas de cuadrantes para k=3.** SPSS 20.

Igualmente, es preciso comentar si renovamos el caso focal, es decir, consideramos como caso focal a otra Corte Superior de Justicia se puede realizar un análisis de medias análogo a lo descrito en el caso focal anterior.

4.3.4.6.- Error del Modelo o de clasificación (Resumen de error) para k=3 vecinos más próximos

El Cuadro 5, muestra un resumen de los errores asociado con el modelo, es decir, es el porcentaje de Cortes Superiores de Justicia del Perú clasificados incorrectamente. Por lo tanto para la muestra de entrenamiento le corresponde una tasa de error de 12.5%, mientras que para la muestra de reserva la tasa es de 0%.

En consecuencia el modelo encontrado es considera idóneo, esto debido a que la tasa de error encontrado es pequeña, esto es, de cada 10 Cortes Superiores de Justicia Clasificados en uno de los grupos (conglomerados) uno de las Cortes Superiores de Justicia es clasificado erróneamente.

Cuadro 5: **Error del Modelo o de clasificación.** SPSS 20.

Resumen de errores

Partición	Porcentaje de registros clasificados incorrectamente
Entrenamiento	12,5%
Reserva	0,0%

4.3.4.7.- Precisión o Exactitud (Tabla de clasificación) para k=3 vecinos más próximos

El Cuadro 6, muestra la tabla de clasificación cruzada de los valores observados en comparación con los valores pronosticados de los grupos (variable destino), en función de la partición (muestra de entrenamiento y muestra de reserva). Asimismo, se evidencia que respecto a la muestra de entrenamiento la Precisión (índice del porcentaje global correcto del pronóstico) es de 87.5%, mientras que para la muestra de reserva el porcentaje global es de 100%. En

consecuencia se puede afirmar que el modelo para el pronóstico es aceptable, debido a que la tasa de clasificación global del modelo es cercano 100%, esto implica que el modelo encontrado es adecuado.

Cuadro 6: **Precisión** (**Índice global pronosticado).** SPSS 20.

Tabla de clasificación		
Partición	**Observado**	**Pronosticado**
		Porcentaje correcto
Entrenamiento	**Porcentaje global**	87,5%
Reserva	**Porcentaje global**	100,0%

4.3.4.8.- Error cuadrático o Índice de error para k=3 (registro de errores de selección)

La Figura 36, muestra el registro de errores de selección para el modelo encontrado, donde se observa que cuando existen tres vecinos más próximos (k=3) el índice de error o de error cuadrático es de 12%, mientras que para valores de cuatro (k=4) y cinco (k=5) vecinos más próximos el índice de error o de error cuadrático es superior al 20%.

En consecuencia se puede afirmar que el modelo para tres (k=3) vecinos más próximo es el más adecuado debido a que tiene el índice de error o de error cuadrático menor comparado con los valores de cuatro (k=4) y cinco (k=5) vecinos más próximos. Por tanto el valor tres 3 sería el valor a posteriori de k.

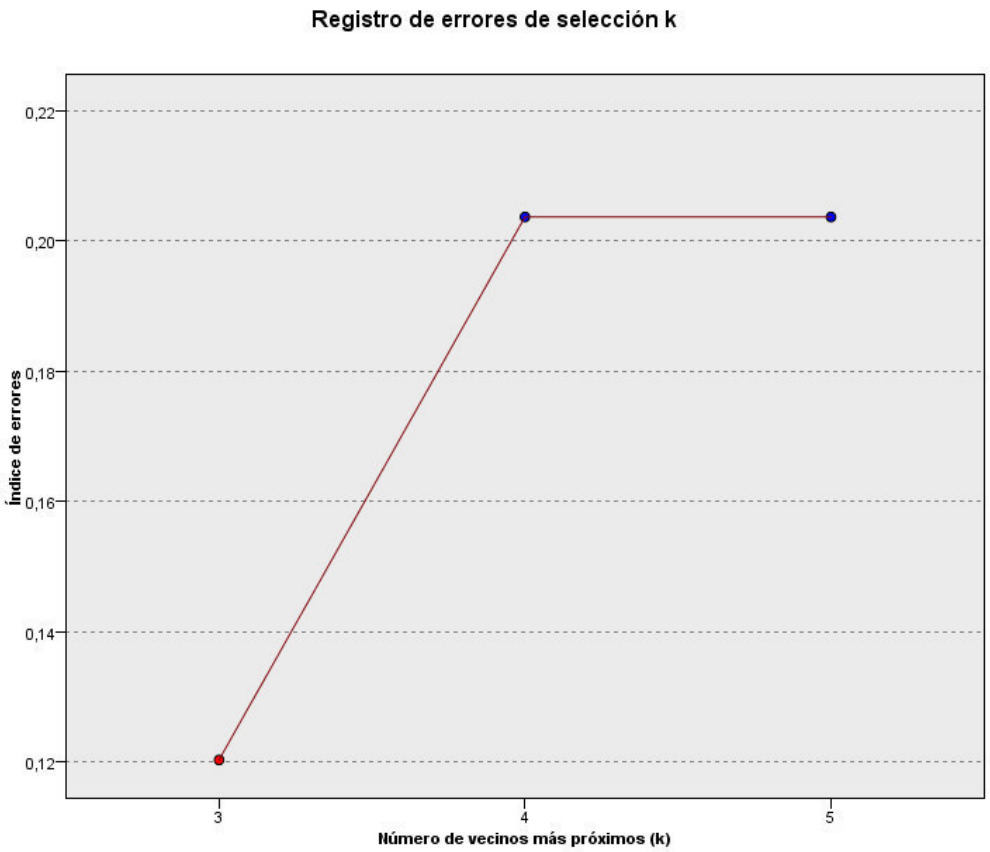


Figura 36: Error cuadrático o Índice de error **(Registro de errores de selección) de k.** SPSS 20.

4.3.4.9 Clasificación 3 vecinos más próximos (3-NN)

El Cuadro 7 muestra los grupos iniciales mediante conglomerados jerárquicos, los grupos pronosticados mediante tres (k=3) vecinos más próximos (3-VMP), las particiones (3-VMP), los pliegues (3-VMP) y las probabilidades (3-VMP) de clasificación para cada grupo.

Cuadro 7: **Grupos y probabilidades para k=3.** SPSS 20.

Corte Superior	Grupos	Grupos Pronosticado VMP	Particiones VMP	Pliegues VMP	Probabilidad 1 - VMP	Probabilidad 2 - VMP	Probabilidad 3 - VMP
Amazonas	1	1	1	2	0.67	0.17	0.17
Ancash	2	2	1	2	0.17	0.67	0.17
Apurímac	1	1	1	2	0.67	0.17	0.17
Cañete	1	1	1	3	0.67	0.17	0.17
Huancave	1	1	1	1	0.67	0.17	0.17
Huaura	2	2	1	1	0.17	0.67	0.17
MadreDios	1	1	1	2	0.67	0.17	0.17
Moquegua	1	1	1	2	0.67	0.17	0.17
Pasco	1	1	1	2	0.67	0.17	0.17
Santa	2	2	1	1	0.17	0.67	0.17
Sullana	1	1	1	3	0.67	0.17	0.17
Tacna	1	1	1	1	0.67	0.17	0.17
Tumbes	1	1	1	1	0.67	0.17	0.17
Ucayali	1	1	1	1	0.67	0.17	0.17
Arequipa	3	3	0	0	0.17	0.17	0.67
Cusco	3	3	0	0	0.17	0.33	0.50
Junín	3	3	1	3	0.17	0.33	0.50
LimaSur	3	2	1	3	0.17	0.67	0.17
Piura	3	3	0	0	0.17	0.33	0.50
Puno	3	2	1	3	0.17	0.67	0.17
Ayacucho	2	2	1	3	0.17	0.67	0.17
Cajamarca	2	2	0	0	0.17	0.50	0.33
Callao	2	2	1	1	0.17	0.50	0.33
Huánuco	2	2	1	2	0.17	0.67	0.17
Ica	3	2	1	1	0.17	0.67	0.17
Loreto	2	2	0	0	0.33	0.50	0.17
SanMartín	2	2	1	3	0.17	0.67	0.17
La Libertad	3	3	1	3	0.17	0.17	0.67
Lambayeq	3	3	1	1	0.17	0.17	0.67
LimaNorte	3	3	0	0	0.17	0.17	0.67

En el Cuadro 7, si observamos al Grupo pronosticado mediante 3 (k) vecinos más próximos, se evidencia que algunas Cortes Superiores cambiaron de grupo respecto al agrupamiento por conglomerados jerárquicos. Con el propósito de exponer este cambio más significativo se realiza un gráfico de dispersión (Figura 32).

4.3.4.10 Dispersión para 3 vecinos más próximos para los grupos

La Figura N° 37 presenta la gráfica de la variable resueltos versus la variable ingresados en un espacio bidimensional de los valores pronosticado para la agrupación realizada por el método de los 3 vecinos más próximo, donde los puntos representan las Cortes Superiores de Justicia en sus respectivos grupos. De otro lado se observa que los grupos formados (3-NN) son diferentes, es decir, no existen solapamientos entre los grupos (pequeño, mediano y grande). En consecuencia la formación de los grupos por k vecinos más próximos es más significativo respecto del agrupamiento realizado por conglomerados jerárquicos tal como se evidencia en el gráfico (Figura 32).

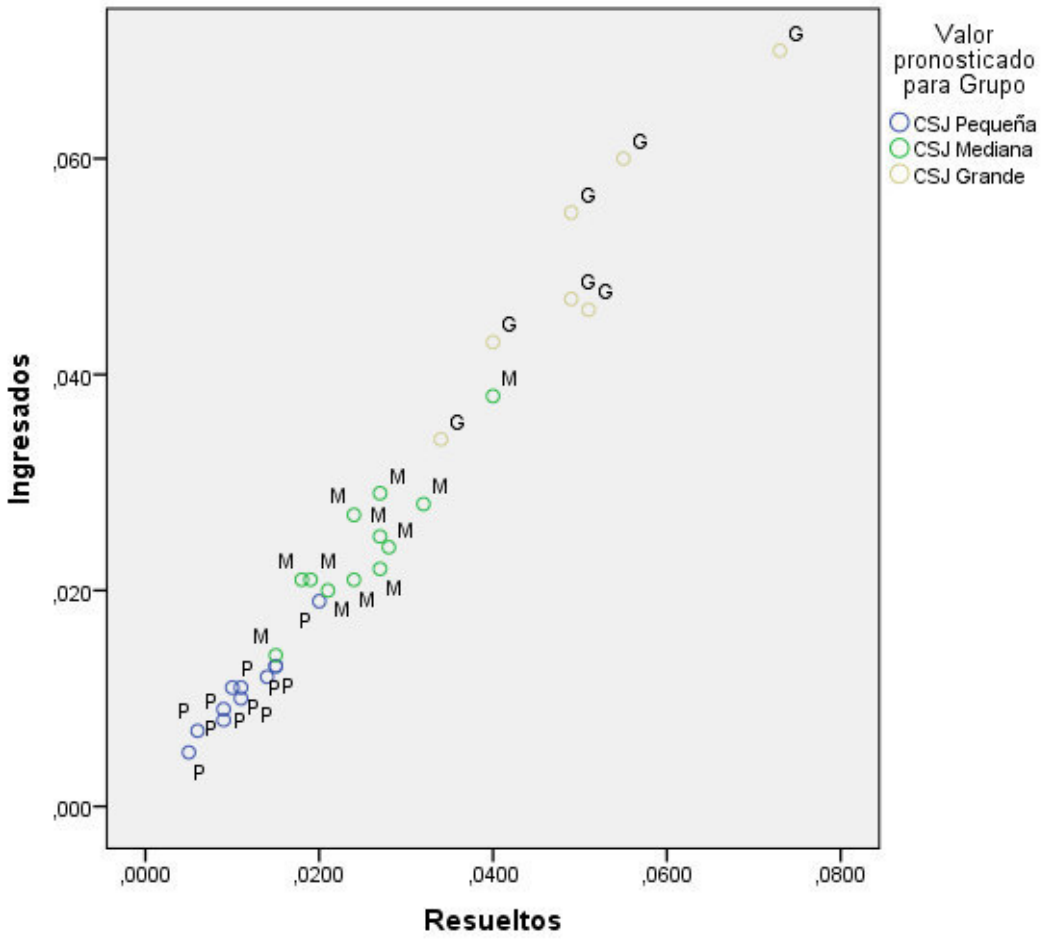


Figura 37: **Dispersión del Modelo 3 vecinos más próximos.** SPSS 20.

4.3.5 Validación del modelo para muestras pequeñas de entrenamiento y reserva.

Para ratificar la precisión del modelo de 3-vecinos más próximos, para muestras pequeñas de entrenamiento y reserva se realiza mediante las pruebas no paramétricas de Kruskal-Wallis y la Prueba de la Mediana.

El Cuadro 8 muestra la prueba estadística no paramétrica de Kruskal-Wallis (Chi-cuadrado), sus grados de libertad (gl) y su nivel crítico (Sig asintótico.) para cada una de las seis variables (Pendiente, Ingresado, Resueltos, Personal, Dependencias y Población). Puesto que el nivel crítico es de 0.000 es menor que 0.05 en cada una de las seis variables, podemos rechazar la hipótesis de igualdad de promedios poblacionales y concluir que los grupos (pequeña, mediana y grande) comparados difieren en cada una de las seis variables Es decir, los grupos son diferentes. Esto también se evidencia en los gráficos de box-plot mostrados en el anexo 7.

Cuadro 8: **Prueba de Kruskal-Wallis: k=3.** SPSS 20.

Estadísticos de contraste[a,b]

	Pendientes	Ingresados	Resueltos	Personal	Dependencias	Población
Chi-cuadrado	24.127	24.868	23.918	24.027	24.168	23.875
gl	2	2	2	2	2	2
Sig. asintót.	.000	.000	.000	.000	.000	.000

a. Prueba de Kruskal-Wallis

b. Variable de agrupación: Valor pronosticado para Grupo

El Cuadro 9 muestra la estadística no paramétrica de Prueba de Mediana, entrega el tamaño de la muestra, el estadístico Chi-cuadrado, sus grados de libertad (gl) y su nivel crítico asintótico (Sig asintótico.) para cada una de las seis variables (Pendiente, Ingresado, Resueltos, Personal, Dependencias y Población). Puesto que el nivel crítico es de 0.000 es menor que 0.05 en cada una de

las seis variables, podemos rechazar la hipótesis de igualdad de Medianas poblacionales y concluir que los grupos (pequeña, mediana y grande) comparados difieren en cada una de las seis variables Es decir, los grupos son diferentes. Esto también se evidencia en los gráficos de box-plot mostrados en el anexo 8.

Cuadro 9: **Prueba de la mediana: k=3.** SPSS 20.

Estadísticos de contraste[a]

	Pendientes	Ingresados	Resueltos	Personal	Dependencias	Población
N	30	30	30	30	30	30
Mediana	.01950	.02100	.022500	.02000	.02600	.02550
Chi-cuadrado	19,333[b]	18,281[c]	19,333[b]	19,333[b]	19,333[b]	19,333[b]
gl	2	2	2	2	2	2
Sig. asintót.	.000	.000	.000	.000	.000	.000

a. Variable de agrupación: Valor pronosticado para Grupo

b. 2 casillas (33,3%) tienen frecuencias esperadas menores que 5. La frecuencia de casilla esperada mínima es 3,5.

c. 2 casillas (33,3%) tienen frecuencias esperadas menores que 5. La frecuencia de casilla esperada mínima es 3,3.

Apoyados en estas pruebas estadísticas de Kruskal-Wallis y la Mediana podemos ejecutar el modelo encontrado de 3 vecinos más próximos con la seguridad que la predicción y estimar se realizarán con precisión. Esto evidencia que el modelo de 3 vecinos más próximos es ejecutable para tamaño datos de entrenamiento y reserva pequeños.

4.4 Modelo de *Predicción mediante k=3 vecinos más próximos*

Esta simulación de predicción es posible debido a que la Corte de Lima no está considerada en el modelo de la presente tesis, debido a que las características de la información que ostenta son muy diferentes respecto a la magnitud de sus variables de las demás Cortes Superiores de Justicia del País.

Para realizar una simulación de predicción se utiliza las variables de la Corte Superior de Lima que está representado mediante un vector (Y) de seis dimensiones que contienen los valores de cada una de las variables.

Y= (pendientes, ingresos, resueltos, personal, dependencias, población)

Y= (0.30, 0.24, 0.22, 0.25, 0.20, 0.16)

Tesis = Matriz (ver Anexo 5)

K = 3

Desarrollando en el modelo:

knn(X, Y, C, k = 3, prob=TRUE) ***…. (1)***

Para valores de:

Y<-matrix(c(0.30,0.24,0.22,0.25,0.20,0.16),nrow=1)

X<-Tesis[2:7]

C<-Tesis[,8]

Se obtiene.

Cuadro 10. **Modelo de predicción para k=3.** R studio.

Modelo	Predicción
knn(X, Y, C, k = 3, prob=TRUE)	Grupo Grande

Si observamos el Cuadro N° 10 se evidencia que la Corte de Lima ha sido clasificada en el grupo tres (Grupo Grande), este resultado corresponde al modelo de 3 vecinos más próximos. En consecuencia este modelo sirve para clasificar futuras Corte Superior de Justicia, cuando utilizamos la función del modelo encontrado (1).

CONCLUSIONES

- La Corte Superior de Lima es excluido para encontrar los modelos de clasificación y predicción del presente estudio, dado que, las características de información que ostenta, son muy diferentes en cuanto a la magnitud de las variables, comparada con las demás Cortes Superiores de Justicia (Figura 28). Además, Lima simboliza un valor extremo (Figura 29), es decir, es numéricamente distante del resto de las Cortes Superiores, en cada una de las variables en estudio. Todo esto apoyado en el análisis descriptivo de datos.

- Se constituyó un modelo de tres grupos (Cuadro 1); Pequeño (n_1=14), Mediano (n_2=9) y Grande (n_3=10), el modelo se sustenta en el método de conglomerados jerárquico mediante encadenamiento simple (vecino más próximo) que construye el árbol jerárquico (Dendrograma) mediante disimilaridad de las Cortes Superiores de Justicia, en un espacio de seis variables.

- Se encontró el modelo óptimo de clasificación y predicción cuando el valor de k es 3 vecinos más próximos, debido a que el error cuadrático (registro de errores de selección) para tres vecinos es 0.12% mientras que para 4 y 5 vecinos es mayor al 20%, evidenciando que el modelo construido para 3 vecinos es más eficiente (Figura 36).

- El modelo pronosticado (3-vecinos más próximos) es preciso o exacto debido a que tiene como índice global de pronóstico (precisión del modelo) para la muestra de reserva de 100% y para la muestra de entrenamiento de 87.5% (Cuadro 6), esto se robustece con el error del modelo (resumen de errores. Cuadro 5) para reserva es de 0% y para entrenamiento es de 12.5%. De otro lado el modelo verifica que la variable más importante es pendientes seguidos de población, dependencia, personal, resueltos e ingresados respectivamente (Figura 34)

- El modelo de 3 vecinos más próximos encontrado se ejecuta con precisión para tamaño muestra de datos de entrenamiento y reserva pequeñas. Esto debido a que los grupos son distintos. Se demuestra mediante las pruebas estadísticas no paramétricas de Kruskal-Wallis y la Mediana, en ambas pruebas rechazamos la hipótesis de igualdad de promedios y medianas poblacionales respectivamente, y concluimos que los grupos (pequeña, mediana y grande) comparados difieren en cada una de las seis variables. Por tanto los grupos son distintos.

- El modelo de tres vecinos más próximos desarrolla el algoritmo que predice la clasificación de futuras Cortes Superiores de Justicia (Cuadro 11). Donde se evidencia que si por ejemplo clasificamos a la Corte Superior de Justicia de Lima, esta se encontraría formando el grupo de Cortes grandes.

RECOMENDACIONES

- Se recomienda crear reflexión en el Poder Judicial sobre los beneficios e importancia de los modelos construidos mediante el método de los vecinos más próximos que serán útiles para mejorar la gestión administrativa en la institución, de otro lado es importante generar un debate académico en la institución sobre lo conveniente del desarrollo y aplicación de este método de clasificación no paramétrica.

- Se recomienda aplicar el modelo de k- vecinos más próximos en otras áreas de gestión del Poder Judicial donde la muestra de entrenamiento y reserva sea grande con el objeto de comprobar que la estimación y predicción del modelo no depende del tamaño de la muestra, es más utilizar otras técnicas de clasificación con el fin de habituar la utilización de esta técnicas estadísticas que permitirán tener una visión más clara a la hora de solucionar los problema de gestión que se presentan en la institución.

- La importancia de las variables (Figura 34) en este modelo sólo está relacionado con la importancia de cada variable para realizar un pronóstico en este estudio, independientemente de si este pronóstico es preciso o no. Además permite considerar eliminar o ignorar las variables que importan menos. Todas estas consideraciones se deben tener presenta al momento de tomar decisiones en la gestión administrativa del Poder Judicial.

- Si bien es cierto al momento de utilizar el modelo predictivo para clasificar futuras Corte Superior de Justicia Lima resulto clasificado como Corte Superior de Justicia Grande, se recomienda considerarlo como un grupo único, es decir, la Corte Superior de Justicia de Lima integre el cuarto grupo, una medida más extrema pero beneficiosa seria la división de la Corte Superior de Lima en dos o tres Cortes Superiores Justicia más pequeño con el fin de mejorar la gestión pública.

REFERENCIA BIBLIOGRÁFICAS

Abellanas M. (1993). *Sobre la vecindad geométrica*. España: Universidad Politécnica de Madrid.

Anderberg, G. M. R. (1973). *Cluster Analysis for Applications*, New York: Academic Press.

Breiman, L., Friedman, J., Olshen, R., & STONE, C. (1984), *Classification and Regresion Trees*, Wadsworth International Group.

Cortijo, F. J. (2001). *Aproximación no paramétrica*, Uruguay: Universidad de la República Uruguay.

Cortijo, F.J. (1995), *Un estudio comparativo de métodos de clasificación de imágenes multibanda*. Tesis Doctoral. Universidad de Granada.

Cortijo, F.J. y Pérez de la Blanca, N. (1997), *A comparative study of some non-parametric spectral classifiers. Applications to problems with high-overlapping training sets.* En International Journal of remote Sensing, vol. 18 (6), págs. 1259-1275.

Cost, S. & Salzberg S. (1993). A weighted nearest neighbor algorithm for learning with symbolic features, *Machine Learning (pp.* 57-78), Boston: Kluwer Academic Publishers.

Cuadras, C. M. (1991). *Métodos de Análisis Multivariante* (2ª Ed.), España: Editorial Universitaria de Barcelona.

Cuadras, C. M. (2012). *Nuevos Métodos de Análisis Multivariante*, España: CMC Editions Manacor Barcelona.

Dasarathy B. V. (1991). *Nearest Neighbour (NN) Norms: NN Pattern Recognition Techniques*, USA: IEEE Computer Society Press.

Devijver, P.A. & Kittler, J.V. (1982), *Pattern Recognition. A Statistical Approach*, Prentice Hall- Englewood Cliffs.

Everitt, B. S. (1993), *Cluster Analysis*, USA: Oxford University Press.

Geva, S. y Sitte, J. (1991), *Adaptative Nearest Neighbor Pattern Classification,* En IEEE Transactions on Neural Networks, vol. 2 (2), págs. 318-322.

Hernández, W. (2007). *13 mitos sobre la carga procesal. Anotaciones y datos para la política judicial pendiente en la materia*. Perú. Justicia Viva Lima.

Huamanchumo, L. E. (2005). *Estandarización de la Carga Procesal Del Poder Judicial de la República del Perú: un enfoque factorial estructurado*. Perú. TECNIA. Vol. 15, 1. Pp. 41-49.

Jean-Philippe L. (2003), Predictors tutorial, Bioinformatic Department Projects.

Jhonson, R. A. & Wichern, D. (1992). *Applied Multivarte Statistical Analysis* (3ª Ed.), London: The Prentice Hall International Editions.

Juárez, C.A. (2004), *Fusión de datos: Imputación y Validación*. Tesis Doctoral. Universidad Politécnica de Cataluña.

Luisa, A. & Oncina, J. (2004). Clasificadores eficaces basados en algoritmos rápidos de búsqueda del vecino más próximo. Memoria para optar el Título de Ingeniero de Sistemas, Departamento de Lenguajes y Sistemas Informáticos. Universidad de Alicante, España.

Mardia, K. V., Kent, J.T. & Bibby, J. M. (1979). *Multivariate Analysis*, New York: Academic Press.

Micó L. & Oncina J. (1998). Comparison of fast nearest neighbour classiers for handwritten character recognition. *Pattern Recognition Letters* (pp. 351-356), Spain: Universidad de Alicante.

Micó, L. (1996). *Algoritmos para la búsqueda del vecino más próximo en espacios métricos.* Tesis doctoral. Universidad Politécnica de Valencia, España.

Morales, G., Mora, J. & Vargas, H. (2008). *Estrategia de regresión basada en el método de los k vecinos más próximos para la estimación de la distancia de falla en sistemas radiales*. Colombia. Rev. Fac. Ing. Univ. Antioquia N.° 45 pp. 100-108.

Morrison, B. F. (1976). *Multivariate Statistical Methods* (2ª Ed.), New York: Academic Press.

Peña, D. (2002). *Análisis de datos Multivariante*, España: Mc Grau-Hill interamericana de España.

Rodríguez, J., Rojas E. & Franco, R. Clasificación de datos usando el método k-nn. Colombia. I+D investigación y desarrollo.

Salas, J. L. (2003). *Bases para la racionalización de la carga procesal: justicia en el reparto de la tarea de administrar justicia*. Perú. Academia de la Magistratura Lima.

Spath, H. & Bull, U. (1980). *Cluster Analysis of Algorithms for Data Reduction and Classifications of Objects*, New York: Wiley.

Venables, W. N. & Ripley, B. D. (1987). *Modern Applied Statistics With s-PLUS* (2ª Ed.), USA: Springer.

Enlaces Web

- http://iie.fing.edu.uy/ense/asign/recpat/material/tema3_00-01/node5.html.
- http://analisisydecision.es/manual-curso-introduccion-de-r-capitulo-17-analisis-cluster-con-r-y-iii/
- http://cran.r-project.org/web/packages/class/class.pdf
- https://rstudio-pubs-static.s3.amazonaws.com/45690_2521251033a04c0da4a011e59e9ca5a1.html
- http://www.ulb.ac.be/assoc/presta/Cursos/Clasi/CLASI_01.PDF
- http://www.ulb.ac.be/assoc/presta/Cursos/Clasi/CLASI_02.PDF
- http://es.slideshare.net/caroman/ia2-algoritmos-clasificacion-vecindad
- http://iie.fing.edu.uy/ense/asign/recpat/material/tema3_00-01/node6.html
- http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t9s.pdf
- http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/t9knn.pdf
- https://es.wikipedia.org/wiki/Validaci%C3%B3n_cruzada#cite_note-loocv-8

Anexos 1: **Variables de las Cortes Superiores de justicia del País.**

N°	CorteSuperior	Pendientes	Ingresados	Resueltos	Personal	Dependencias	Población
1	Amazonas	0.007	0.011	0.011	0.017	0.018	0.014
2	Ancash	0.020	0.020	0.021	0.017	0.029	0.020
3	Apurímac	0.013	0.013	0.015	0.016	0.017	0.015
4	Cañete	0.006	0.008	0.009	0.015	0.013	0.008
5	Huancavelica	0.005	0.009	0.009	0.009	0.010	0.010
6	Huaura	0.017	0.021	0.019	0.022	0.024	0.019
7	MadreDios	0.004	0.007	0.006	0.009	0.012	0.004
8	Moquegua	0.007	0.011	0.010	0.012	0.012	0.006
9	Pasco	0.006	0.005	0.005	0.006	0.009	0.007
10	Santa	0.026	0.022	0.027	0.024	0.031	0.018
11	Sullana	0.013	0.012	0.014	0.011	0.013	0.018
12	Tacna	0.014	0.019	0.020	0.017	0.016	0.011
13	Tumbes	0.008	0.010	0.011	0.014	0.012	0.008
14	Ucayali	0.012	0.013	0.015	0.015	0.017	0.017
15	Arequipa	0.057	0.055	0.049	0.055	0.048	0.041
16	Cusco	0.026	0.043	0.040	0.044	0.048	0.042
17	Junín	0.047	0.047	0.049	0.037	0.038	0.051
18	LimaSur	0.035	0.025	0.027	0.017	0.019	0.049
19	Piura	0.041	0.034	0.034	0.034	0.031	0.042
20	Puno	0.013	0.021	0.018	0.029	0.034	0.046
21	Ayacucho	0.020	0.024	0.028	0.018	0.025	0.025
22	Cajamarca	0.022	0.027	0.024	0.031	0.034	0.034
23	Callao	0.032	0.028	0.032	0.037	0.032	0.032
24	Huánuco	0.023	0.021	0.024	0.021	0.027	0.026
25	Ica	0.026	0.038	0.040	0.039	0.039	0.026
26	Loreto	0.015	0.014	0.015	0.019	0.022	0.030
27	SanMartín	0.019	0.029	0.027	0.024	0.029	0.031
28	La Libertad	0.052	0.070	0.073	0.051	0.051	0.059
29	Lambayeque	0.054	0.060	0.055	0.047	0.050	0.057
30	LimaNorte	0.056	0.046	0.051	0.041	0.042	0.079
31	Lima	0.301	0.239	0.221	0.250	0.198	0.156

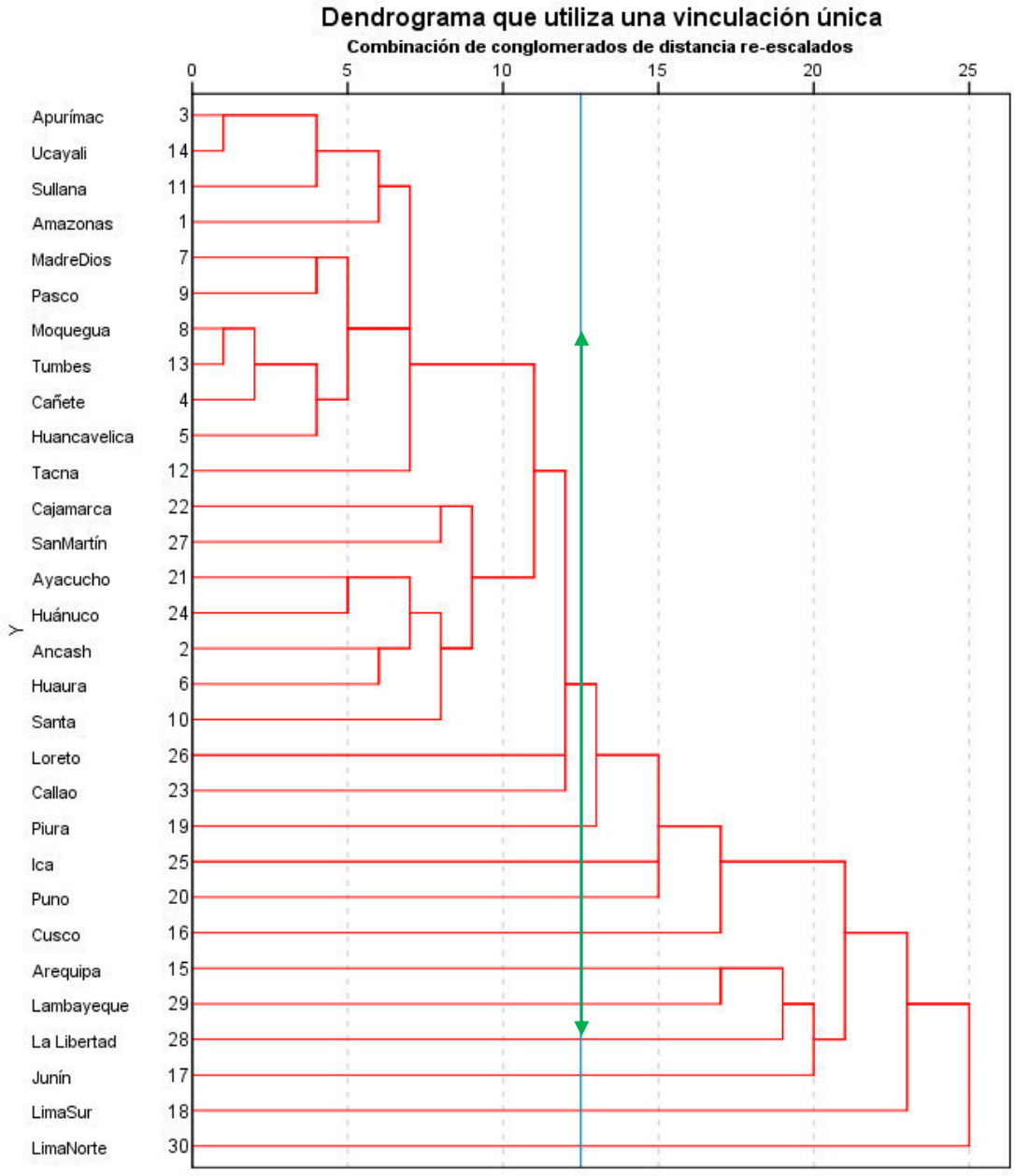
Anexo 2: **Distancia Euclidiana**.

Caso	Distancia euclídea																														
	1: Amazonas	2: Ancash	3: Apurímac	4: Arequipa	5: Ayacucho	6: Cajamarca	7: Callao	8:Cañete	9:Cusco	10:Huancavelica	11:Huánuco	12:Huaura	13:Ica	14:Junín	15:La Libertad	16:Lambayeque	17:Lima	18:LimaNorte	19:LimaSur	20:Loreto	21:MadreDios	22:Moquegua	23: Pasco	24: Piura	25: Puno	26: SanMartín	27: Santa	28: Sullana	29: Tacna	30: Tumbes	31: Ucayali
1:Amazonas	0	188439	35554	832406	322955	613416	553309	166831	840142	112959	361819	152883	368904	1118356	1361567	1315586	4321673	1976726	1055547	472422	289968	242650	209642	848581	959737	507014	141394	118086	90045	189285	92562
2:Ancash	188439	0	153349	644001	134811	425647	365094	354839	652179	301146	173736	35789	180805	930196	1173352	1127408	4133244	1788858	867815	285344	477920	430379	397716	660477	772511	319155	50302	70818	275844	377054	95944
3:Apurímac	35554	153349	0	797326	288046	578728	518389	201523	805390	147816	326942	117833	333823	1083494	1326672	1280719	4286516	1942020	1020877	437919	324644	277183	244389	813734	925285	472283	105901	83417	123269	223816	57808
4:Arequipa	832406	644001	797326	0	509672	222956	279898	998710	42958	945097	470966	679572	464128	288398	530446	484679	3489332	1147222	232176	363710	1121659	1074016	1041582	38952	150861	327025	692281	714492	919181	1020794	739880
5:Ayacucho	322955	134811	288046	509672	0	291072	230540	489531	517422	435776	39664	170356	48678	795526	1038641	992739	3998967	1654207	733273	151773	612626	565075	532424	525892	638073	184480	184049	205058	410482	511762	230432
6:Cajamarca	613416	425647	578728	222956	291072	0	62893	780193	227131	726363	251976	461166	247966	505234	748782	702467	3709925	1363346	442319	141721	903331	855872	823019	235604	347192	106723	474960	495438	701440	802525	520982
7:Callao	553309	365094	518389	279898	230540	62893	0	719910	287571	666173	191496	400725	186712	565132	808497	762388	3768848	1423875	503127	85449	843007	795470	762765	295424	408972	48587	413998	435268	640837	742145	460763
8:Cañete	166831	354839	201523	998710	489531	780193	719910	0	1006859	53928	528463	319261	535021	1285012	1528141	1482226	4487749	2143502	1222326	639232	123150	75959	42974	1015252	1126564	673751	306578	284787	80630	22789	259222
9:Cusco	840142	652179	805390	42958	517422	227131	287571	1006859	0	953045	478587	687686	472894	278776	521819	475770	3483485	1137109	217842	368776	1129966	1082430	1049732	23116	126626	333142	701192	722210	927822	1029126	747686
10:Huancavelica	112959	301146	147816	945097	435776	726363	666173	53928	953045	0	474701	265553	481408	1231249	1474399	1428469	4434257	2089679	1168501	585375	177023	129799	96753	961489	1072680	619931	253179	230989	31287	76500	205427
11:Huánuco	361819	173736	326942	470966	39664	251976	191496	528463	478587	474701	0	209355	25546	756569	999879	953822	3960207	1615130	694090	112595	651583	604080	571314	486806	598951	145627	223062	243780	449573	550736	269279
12:Huaura	152883	35789	117833	679572	170356	461166	400725	319261	687686	265553	209355	0	216136	965786	1208902	1162977	4168824	1824438	903390	320763	442335	394788	362148	696083	807969	354642	**19375**	36139	240288	341477	60575
13:Ica	368904	180805	333823	464128	48678	247966	186712	535021	472894	481408	25546	216136	0	750738	993382	947776	3953171	1609787	689465	113744	657946	610271	577958	481562	594923	141706	228883	251472	455433	557078	276553
14:Junín	1118356	930196	1083494	288398	795526	505234	565132	1285012	278776	1231249	756569	965786	750738	0	244209	197366	3205083	859214	71638	646782	1408111	1360557	1327872	269947	167680	611453	978991	1000332	1205864	1307247	1025836
15:La Libertad	1361567	1173352	1326672	530446	1038641	748782	808497	1528141	521819	1474399	999879	1208902	993382	244209	0	50235	2962121	618688	311250	890437	1651189	1603571	1571043	513841	408428	854722	1221973	1243618	1448781	1550313	1269056
16:Lambayeque	1315586	1127408	1280719	484679	992739	702467	762388	1482226	475770	1428469	953822	1162977	947776	197366	50235	0	3008141	662791	263348	844035	1605309	1557736	1525092	467232	361124	808663	1176148	1197582	1403011	1504443	1223072
17:Lima	4321673	4133244	4286516	3489332	3998967	3709925	3768848	4487749	3483485	4434257	3960207	4168824	3953171	3205083	2962121	3008141	0	2354525	3270256	3851474	4610513	4562822	4530594	3474447	3368103	3815719	4181198	4203766	4407960	4509708	4229153
18:LimaNorte	1976726	1788858	1942020	1147222	1654207	1363346	1423875	2143502	1137109	2089679	1615130	1824438	1609787	859214	618688	662791	2354525	0	921248	1504429	2266644	2219187	2186309	1128510	1017726	1469889	1837872	1858718	2064663	2165834	1884287
19:LimaSur	1055547	867815	1020877	232176	733273	442319	503127	1222326	217842	1168501	694090	903390	689465	71638	311250	263348	3270256	921248	0	583220	1345468	1298055	1265108	208460	99314	548983	916999	937542	1143655	1244686	963134
20:Loreto	472422	285344	437919	363710	151773	141721	85449	639232	368776	585375	112595	320763	113744	646782	890437	844035	3851474	1504429	583220	0	762371	715028	681992	377053	487401	39938	335108	354522	560929	661652	380134
21:MadreDios	289968	477920	324644	1121659	612626	903331	843007	123150	1129966	177023	651583	442335	657946	1408111	1651189	1605309	4610513	2266644	1345468	762371	0	47804	80449	1138363	1249691	796874	429441	407933	202659	100885	382365
22:Moquegua	242650	430379	277183	1074016	565075	855872	795470	75959	1082430	129799	604080	394788	610271	1360557	1603571	1557736	4562822	2219187	1298055	715028	47804	0	34314	1090849	1202362	749373	381797	360538	154912	53401	334935
23:Pasco	209642	397716	244389	1041582	532424	823019	762765	42974	1049732	96753	571314	362148	577958	1327872	1571043	1525092	4530594	2186309	1265108	681992	80449	34314	0	1058085	1169315	716611	349460	327602	123308	21908	302070
24:Piura	848581	660477	813734	38952	525892	235604	295424	1015252	23116	961489	486806	696083	481562	269947	513841	467232	3474447	1128510	208460	377053	1138363	1090849	1058085	0	119951	341862	709391	730533	936239	1037516	756067
25:Puno	959737	772511	925285	150861	638073	347192	408972	1126564	126626	1072680	598951	807969	594923	167680	408428	361124	3368103	1017726	99314	487401	1249691	1202362	1169315	119951	0	453758	822008	841900	1048211	1148999	867488
26:SanMartín	507014	319155	472283	327025	184480	106723	48587	673751	333142	619931	145627	354642	141706	611453	854722	808663	3815719	1469889	548983	39938	796874	749373	716611	341862	453758	0	368446	389073	594873	696047	414559
27:Santa	141394	50302	105901	692281	184049	474960	413998	306578	701192	253179	223062	19375	228883	978991	1221973	1176148	4181198	1837872	916999	335108	429441	381797	349460	709391	822008	368446	0	30990	227011	328576	51165
28:Sullana	118086	70818	83417	714492	205058	495438	435268	284787	722210	230989	243780	36139	251472	1000332	1243618	1197582	4203766	1858718	937542	354522	407933	360538	327602	730533	841900	389073	30990	0	206613	307158	25710
29:Tacna	90045	275844	123269	919181	410482	701440	640837	80630	927822	31287	449573	240288	455433	1205864	1448781	1403011	4407960	2064663	1143655	560929	202659	154912	123308	936239	1048211	594873	227011	206613	0	101952	180955
30:Tumbes	189285	377054	223816	1020794	511762	802525	742145	22789	1029126	76500	550736	341477	557078	1307247	1550313	1504443	4509708	2165834	1244686	661652	100885	53401	**21908**	1037516	1148999	696047	328576	307158	101952	0	281566
31:Ucayali	92562	95944	57808	739880	230432	520982	460763	259222	747686	205427	269279	60575	276553	1025836	1269056	1223072	4229153	1884287	963134	380134	382365	334935	302070	756067	867488	414559	51165	25710	180955	281566	0

Anexo 3: **Historial de conglomeración aplicando vecinos más próximos.**

Historial de conglomeración						
Etapa	Conglomerado que se combina		Coeficientes	Etapa en la que el conglomerado aparece por primera vez		Próxima etapa
	Conglomerado 1	Conglomerado 2		Conglomerado 1	Conglomerado 2	
1	3	14	.002	0	0	4
2	8	13	.003	0	0	3
3	4	8	.004	0	2	6
4	3	11	.006	1	0	9
5	7	9	.006	0	0	8
6	4	5	.006	3	0	8
7	21	24	.007	0	0	11
8	4	7	.007	6	5	12
9	1	3	.008	0	4	12
10	2	6	.008	0	0	11
11	2	21	.009	10	7	15
12	1	4	.009	9	8	13
13	1	12	.009	12	0	17
14	22	27	.010	0	0	16
15	2	10	.010	11	0	16
16	2	22	.011	15	14	17
17	1	2	.013	13	16	18
18	1	26	.014	17	0	19
19	1	23	.014	18	0	20
20	1	19	.015	19	0	21
21	1	25	.017	20	0	22
22	1	20	.017	21	0	23
23	1	16	.020	22	0	27
24	15	29	.020	0	0	25
25	15	28	.021	24	0	26
26	15	17	.023	25	0	27
27	1	15	.024	23	26	28
28	1	18	.025	27	0	29
29	1	30	.028	28	0	0

Anexo 4: **Árbol jerárquico o Dendograma**



Anexo 5: **Matriz: De variables y grupo pronosticado mediante 3-vecinos más próximos.**

CorteSuperior	Pendientes	Ingresados	Resueltos	Personal	Dependencias	Población	Grupo
Amazonas	0.01	0.01	0.01	0.02	0.02	0.01	1
Ancash	0.02	0.02	0.02	0.02	0.03	0.02	2
Apurímac	0.01	0.01	0.02	0.02	0.02	0.02	1
Cañete	0.01	0.01	0.01	0.02	0.01	0.01	1
Huancave	0.01	0.01	0.01	0.01	0.01	0.01	1
Huaura	0.02	0.02	0.02	0.02	0.02	0.02	2
MadreDio	0.00	0.01	0.01	0.01	0.01	0.00	1
Moquegua	0.01	0.01	0.01	0.01	0.01	0.01	1
Pasco	0.01	0.01	0.01	0.01	0.01	0.01	1
Santa	0.03	0.02	0.03	0.02	0.03	0.02	2
Sullana	0.01	0.01	0.01	0.01	0.01	0.02	1
Tacna	0.01	0.02	0.02	0.02	0.02	0.01	1
Tumbes	0.01	0.01	0.01	0.01	0.01	0.01	1
Ucayali	0.01	0.01	0.02	0.02	0.02	0.02	1
Arequipa	0.06	0.06	0.05	0.06	0.05	0.04	3
Cusco	0.03	0.04	0.04	0.04	0.05	0.04	3
Junín	0.05	0.05	0.05	0.04	0.04	0.05	3
LimaSur	0.04	0.03	0.03	0.02	0.02	0.05	3
Piura	0.04	0.03	0.03	0.03	0.03	0.04	3
Puno	0.01	0.02	0.02	0.03	0.03	0.05	3
Ayacucho	0.02	0.02	0.03	0.02	0.03	0.03	2
Cajamarc	0.02	0.03	0.02	0.03	0.03	0.03	2
Callao	0.03	0.03	0.03	0.04	0.03	0.03	2
Huánuco	0.02	0.02	0.02	0.02	0.03	0.03	2
Ica	0.03	0.04	0.04	0.04	0.04	0.03	3
Loreto	0.02	0.01	0.02	0.02	0.02	0.03	2
SanMartí	0.02	0.03	0.03	0.02	0.03	0.03	2
La Liber	0.05	0.07	0.07	0.05	0.05	0.06	3
Lambayeq	0.05	0.06	0.06	0.05	0.05	0.06	3
LimaNort	0.06	0.05	0.05	0.04	0.04	0.08	3

Anexo 6: **Sintaxis del Modelo k (3) - NN (SPSS 20)**.

```
KNN is available in the Statistics Base option.
KNN [dependent variable [(MLEVEL = {S})]]
                        {O}
                        {N}
   [BY factor-list] [WITH covariate-list]
[/EXCEPT VARIABLES = varlist]
[/CASELABELS VARIABLE = varname]
[/FOCALCASES VARIABLE = varname]
[/RESCALE [COVARIATE = {ADJNORMALIZED**}]]
                {NONE          }
[/PARTITION {TRAINING = {70**   } HOLDOUT = {30**   }}]
                 {integer}          {integer}
         {VARIABLE = varname                  }
[/MODEL [METRIC = {EUCLID** }]
              {CITYBLOCK}
      [NEIGHBORS = {FIXED**} [(K={3**    })                ]]
                         {integer}
                {AUTO   } [(KMIN={3      }, KMAX={5      })]
                           {integer}       {integer}
      [FEATURES = {ALL**}                              ]
                {AUTO } [(FORCE = variable [variable ...])]
[/CRITERIA  [NUMFEATURES = {AUTO**                    }]
                    {FIXED(integer)             }
                    {ERRORRATIO(MINCHANGE={0.01 })}
                                {value}
      [PREDICTED = {MEAN**}]
                {MEDIAN}
      [WEIGHTFEATURES = {NO**}]
                   {YES }
[/CROSSVALIDATION {FOLDS = {10**   } }]
                    {integer}
               {VARIABLE = varname}
[/MISSING USERMISSING = {EXCLUDE**}]
                  {INCLUDE  }
[/VIEWMODEL [DISPLAY = {YES**}]]
                  {NO   }
[/PRINT [CPS**] [NONE]]
[/SAVE [PREDVAL[(varname)]]     ]
      [PREDPROB[(rootname)]]
      [PARTITION[(varname)]]
      [FOLD[(varname)]]
      [MAXCAT({25**   })]
            {integer}

[/OUTFILE [MODEL ='filename']               ]
        [FOCALCASES = 'savfile' | 'dataset'].
```

** Default if the subcommand or keyword is omitted.

Anexo 7: **Sintaxis de los resultados del Modelo k (3) - NN (SPSS 20)**.


```
<?xml version="1.0" encoding="UTF-8" ?>
- <PMML version="4.0" xmlns="http://www.dmg.org/PMML-4_0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.dmg.org/PMML-4_0 pmml-4-0.xsd">
- <Header copyright="Copyright (c) IBM Corp. 1999, 2012.">
<Application name="IBM SPSS Statistics" version="21.0.0.0" />
</Header>
- <MiningBuildTask>
- <Extension extender="spss.com" name="SPSS Model Settings">
- <Parameters>
<Parameter name="CaseIDVar" type="fieldName" value="$casecount" />
<Parameter name="CaseLabelVar" type="fieldName" value="CorteSuperior"
/>
<Parameter name="VFoldVar" type="fieldName" value="$crossvalidation" />
<Parameter name="FocalVar" type="fieldName" value="$focalvariable" />
<Parameter name="RescaleMethod" type="enum" value="true" />
<Parameter name="DistMetric" type="enum" value="Euclidean" />
<Parameter name="ComputePredValFunc" type="enum" value="Mean" />
<Parameter name="use_caselabels" type="boolean" value="true" />
<Parameter name="IsFeatureImportance" type="boolean" value="false" />
<Parameter name="IsFeatureSelection" type="boolean" value="false" />
<Parameter name="IsAutoFeatureSelection" type="boolean" value="false" />
<Parameter name="IsFeatureWeight" type="boolean" value="false" />
<Parameter name="IsAutoKSelection" type="boolean" value="true" />
<Parameter name="NSpecifiedFeatures" type="integer" value="0" />
<Parameter name="MinK" type="integer" value="3" />
<Parameter name="MaxK" type="integer" value="5" />
<Parameter name="VFoldValue" type="integer" value="3" />
<Parameter name="SpecifiedErrRange" type="double" value="0.01" />
<Parameter name="TempFileDir" type="string"
value="C:\Users\nquezada\AppData\Local\Temp\spss3628\" />
- <Parameter name="target">
- <ListValue>
<Value value="Grupo" />
</ListValue>
</Parameter>
- <Parameter name="inputs">
- <ListValue>
<Value value="Pendientes" />
<Value value="Ingresados" />
<Value value="Resueltos" />
<Value value="Personal" />
<Value value="Dependencias" />
<Value value="Población" />
</ListValue>
</Parameter>
</Parameters>
</Extension>
</MiningBuildTask>
- <DataDictionary numberOfFields="11">
```

```
- <DataField name="Grupo" displayName="Grupos" optype="categorical"
    dataType="double">
  <Value value="1" displayValue="CSJ Pequeña" property="valid" />
  <Value value="2" displayValue="CSJ Mediana" property="valid" />
  <Value value="3" displayValue="CSJ Grande" property="valid" />
    </DataField>
- <DataField name="Pendientes" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.004" rightMargin="0.057" />
    </DataField>
- <DataField name="Ingresados" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.005" rightMargin="0.07" />
    </DataField>
- <DataField name="Resueltos" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.005" rightMargin="0.073" />
    </DataField>
- <DataField name="Personal" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.006" rightMargin="0.055" />
    </DataField>
- <DataField name="Dependencias" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.009" rightMargin="0.051" />
    </DataField>
- <DataField name="Población" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0.004" rightMargin="0.079" />
    </DataField>
- <DataField name="CorteSuperior" optype="categorical" dataType="string">
  <Value value="Amazonas" property="valid" />
  <Value value="Ancash" property="valid" />
  <Value value="Apurímac" property="valid" />
  <Value value="Cañete" property="valid" />
  <Value value="Huancave" property="valid" />
  <Value value="Huaura" property="valid" />
  <Value value="MadreDio" property="valid" />
  <Value value="Moquegua" property="valid" />
  <Value value="Pasco" property="valid" />
  <Value value="Santa" property="valid" />
  <Value value="Sullana" property="valid" />
  <Value value="Tacna" property="valid" />
  <Value value="Tumbes" property="valid" />
  <Value value="Ucayali" property="valid" />
  <Value value="Arequipa" property="valid" />
  <Value value="Junín" property="valid" />
  <Value value="LimaSur" property="valid" />
  <Value value="Piura" property="valid" />
  <Value value="Puno" property="valid" />
  <Value value="Ayacucho" property="valid" />
  <Value value="Cajamarc" property="valid" />
  <Value value="Callao" property="valid" />
  <Value value="Huánuco" property="valid" />
  <Value value="Ica" property="valid" />
  <Value value="SanMartí" property="valid" />
  <Value value="La Liber" property="valid" />
  <Value value="Lambayeq" property="valid" />
  <Value value="LimaNort" property="valid" />
```

```
    </DataField>
- <DataField name="$crossvalidation" displayName="Validación cruzada"
    optype="categorical" dataType="double">
  <Value value="0" property="valid" />
  <Value value="1" property="valid" />
  <Value value="2" property="valid" />
    </DataField>
- <DataField name="$casecount" displayName="Recuento de casos"
    optype="continuous" dataType="integer">
      <Interval closure="openClosed" leftMargin="-1.79769e+308"
    rightMargin="1.79769e+308" />
    </DataField>
- <DataField name="$focalvariable" optype="continuous" dataType="double">
  <Interval closure="openClosed" leftMargin="0" rightMargin="1" />
    </DataField>
    </DataDictionary>
- <Extension extender="spss.com" name="DataDictionaryFieldProperties">
- <FieldExtensions>
  <FieldExtension formatName="F8.2" name="Grupo" />
  <FieldExtension formatName="F8.3" name="Pendientes" />
  <FieldExtension formatName="F8.3" name="Ingresados" />
  <FieldExtension formatName="F8.4" name="Resueltos" />
  <FieldExtension formatName="F8.3" name="Personal" />
  <FieldExtension formatName="F8.3" name="Dependencias" />
  <FieldExtension formatName="F8.3" name="Población" />
  <FieldExtension formatName="A24" name="CorteSuperior" />
    </FieldExtensions>
- <FieldFormats>
  <NumberFormat alignment="auto" decimalPlaces="2" decimalSymbol="local"
    formatType="standard" groupingSymbol="local" name="F8.2" width="8"
    />
  <NumberFormat alignment="auto" decimalPlaces="3" decimalSymbol="local"
    formatType="standard" groupingSymbol="local" name="F8.3" width="8"
    />
  <NumberFormat alignment="auto" decimalPlaces="4" decimalSymbol="local"
    formatType="standard" groupingSymbol="local" name="F8.4" width="8"
    />
    <StringFormat alignment="auto" hexadecimal="false" name="A24"
    width="24" />
    </FieldFormats>
    </Extension>
- <Extension extender="spss.com" name="SPSS Models">
- <NearestNeighborModels>
- <SimpleTable name="originalData">
  <RowNames />

    <ColumnNames>ID;Grupo;Pendientes;Ingresados;Resueltos;Personal;Dep
    endencias;Población;CorteSuperior;$crossvalidation;$casecount;$focalvaria
    ble</ColumnNames>
  <Row>1;1;0.007;0.011;0.011;0.017;0.018;0.014;Amazonas;1;1;0</Row>
  <Row>2;2;0.02;0.02;0.021;0.017;0.029;0.02;Ancash;1;2;0</Row>
  <Row>3;1;0.013;0.013;0.015;0.016;0.017;0.015;Apurímac;1;3;0</Row>
  <Row>4;1;0.006;0.008;0.009;0.015;0.013;0.008;Cañete;2;4;0</Row>
  <Row>5;1;0.005;0.009;0.009;0.009;0.01;0.01;Huancave;0;5;0</Row>
```

```
<Row>6;2;0.017;0.021;0.019;0.022;0.024;0.019;Huaura;0;6;0</Row>
<Row>7;1;0.004;0.007;0.006;0.009;0.012;0.004;MadreDio;1;7;0</Row>
<Row>8;1;0.007;0.011;0.01;0.012;0.012;0.006;Moquegua;1;8;0</Row>
<Row>9;1;0.006;0.005;0.005;0.006;0.009;0.007;Pasco;1;9;0</Row>
<Row>10;2;0.026;0.022;0.027;0.024;0.031;0.018;Santa;0;10;0</Row>
<Row>11;1;0.013;0.012;0.014;0.011;0.013;0.018;Sullana;2;11;0</Row>
<Row>12;1;0.014;0.019;0.02;0.017;0.016;0.011;Tacna;0;12;0</Row>
<Row>13;1;0.008;0.01;0.011;0.014;0.012;0.008;Tumbes;0;13;0</Row>
<Row>14;1;0.012;0.013;0.015;0.015;0.017;0.017;Ucayali;0;14;0</Row>
<Row>15;3;0.057;0.055;0.049;0.055;0.048;0.041;Arequipa;2;15;0</Row>
<Row>16;3;0.047;0.047;0.049;0.037;0.038;0.051;Junín;2;17;0</Row>
<Row>17;3;0.035;0.025;0.027;0.017;0.019;0.049;LimaSur;2;18;0</Row>
<Row>18;3;0.041;0.034;0.034;0.034;0.031;0.042;Piura;2;19;0</Row>
<Row>19;3;0.013;0.021;0.018;0.029;0.034;0.046;Puno;0;20;0</Row>
<Row>20;2;0.02;0.024;0.028;0.018;0.025;0.025;Ayacucho;1;21;0</Row>
<Row>21;2;0.022;0.027;0.024;0.031;0.034;0.034;Cajamarc;0;22;0</Row>
<Row>22;2;0.032;0.028;0.032;0.037;0.032;0.032;Callao;2;23;0</Row>
<Row>23;2;0.023;0.021;0.024;0.021;0.027;0.026;Huánuco;2;24;0</Row>
<Row>24;3;0.026;0.038;0.04;0.039;0.039;0.026;Ica;0;25;0</Row>
<Row>25;2;0.019;0.029;0.027;0.024;0.029;0.031;SanMartí;0;27;0</Row>
<Row>26;3;0.052;0.07;0.073;0.051;0.051;0.059;La Liber;2;28;0</Row>
<Row>27;3;0.054;0.06;0.055;0.047;0.05;0.057;Lambayeq;2;29;0</Row>
<Row>28;3;0.056;0.046;0.051;0.041;0.042;0.079;LimaNort;0;30;0</Row>
  </SimpleTable>
- <NearestNeighborModel modelName="Grupo" response="Grupo">
- <SimpleTable name="AnalysisOptions">
  <RowNames />

    <ColumnNames>distanceMetric;continuousTransformation;categoricalTran
    sformationselected;continuousPrediction;K;selectedFeatures</ColumnNam
    es>
<Row>Euclidean;Normalized;one-of-c;Mean;3;6</Row>
  </SimpleTable>
- <MiningSchema>
  <MiningField name="Pendientes" usageType="active" />
  <MiningField name="Ingresados" usageType="active" />
  <MiningField name="Resueltos" usageType="active" />
  <MiningField name="Personal" usageType="active" />
  <MiningField name="Dependencias" usageType="active" />
  <MiningField name="Población" usageType="active" />
  <MiningField name="Grupo" usageType="predicted" />
  </MiningSchema>
- <SimpleTable name="KSelection">
  <RowNames />
  <ColumnNames>K;error</ColumnNames>
  <Row>3;0.096969696969697</Row>
  <Row>4;0.096969696969697</Row>
  <Row>5;0.260606060606061</Row>
  </SimpleTable>
- <SimpleTable name="neighborsDistances">
  <RowNames />
```

<ColumnNames>ID;$casecount;neighbor1;neighbor2;neighbor3;distance1
;distance2;distance3;prediction1;prediction2;prediction3</ColumnNames>

<Row>1;1;14;3;4;0.261837991973809;0.271175708650088;0.31993459
3979916;0;0;0</Row>

<Row>2;2;24;21;6;0.28787847504494;0.336541827525542;0.34098778
3083016;1;1;1</Row>

<Row>3;3;14;12;1;0.077035130037 6704;0.269155078168591;0.271175
708650088;0;0;0</Row>

<Row>4;4;13;8;5;0.129910404836053;0.175828117802529;0.29257197
0612757;0;0;0</Row>

<Row>5;5;8;7;9;0.213985598891996;0.218327136346245;0.232538341
026076;0;0;0</Row>

<Row>6;6;2;24;21;0.340987783083016;0.360289706685993;0.3820065
40293524;1;1;1</Row>

<Row>7;7;5;9;8;0.218327136346245;0.228363585617572;0.244218969
835763;0;0;0</Row>

<Row>8;8;13;4;5;0.112889885550149;0.175828117802529;0.21398559
8891996;0;0;0</Row>

<Row>9;9;7;5;8;0.228363585617572;0.232538341026076;0.371789479
285401;0;0;0</Row>

<Row>10;10;24;2;21;0.343997389930328;0.423957589855048;0.48204
5575661111;1;1;1</Row>

<Row>11;11;14;3;13;0.258618719743868;0.293500067103749;0.36816
4018322617;0;0;0</Row>

<Row>12;12;3;14;11;0.269155078168591;0.309736305222025;0.44066
3151945555;0;0;0</Row>

<Row>13;13;8;4;5;0.112889885550149;0.129910404836053;0.2660581
77709605;0;0;0</Row>

<Row>14;14;3;11;1;0.077035130037 6704;0.258618719743868;0.26183
7991973809;0;0;0</Row>

<Row>15;15;29;28;17;0.604453005835543;1.01210238055357;1.02011
505273434;2;2;2</Row>

<Row>16;17;19;30;29;0.767857627988226;0.860347803061717;0.8829
75340991741;2;2;2</Row>

<Row>17;18;24;21;27;0.880878328951118;0.902833950094395;0.9583
73879808738;1;1;1</Row>

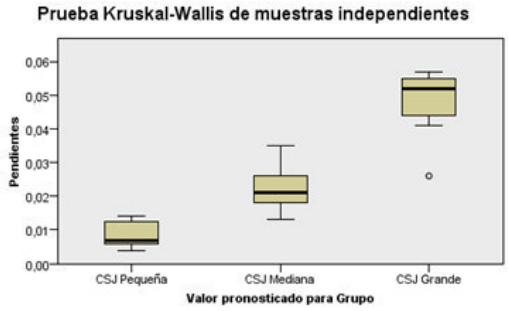
<Row>18;19;23;17;22;0.49118125359721;0.767857627988226;0.85315
338221654;1;2;1</Row>

```
<Row>19;20;22;27;24;0.538174820500232;0.663523451178154;0.8220
21347169873;1;1;1</Row>

<Row>20;21;24;2;6;0.244852297812329;0.336541827525542;0.382006
540293524;1;1;1</Row>

<Row>21;22;27;23;20;0.411230341265824;0.520191548763597;0.5381
74820500232;1;1;1</Row>

<Row>22;23;19;22;25;0.49118125359721;0.520191548763597;0.58709
1708435804;2;2;2</Row>

<Row>23;24;21;2;10;0.244852297812329;0.28787847504494;0.343997
389930328;1;1;1</Row>

<Row>24;25;23;22;19;0.587091708435804;0.753403015209732;0.8576
17041397494;1;1;1</Row>

<Row>25;27;24;21;22;0.364695650007186;0.384465700435086;0.4112
30341265824;1;1;1</Row>

<Row>26;28;29;15;30;0.642195397337876;1.01210238055357;1.27338
511014389;2;2;2</Row>

<Row>27;29;15;28;30;0.604453005835543;0.642195397337876;0.8685
48468760855;2;2;2</Row>

<Row>28;30;17;29;15;0.860347803061717;0.868548468760855;1.2314
9433705543;2;2;2</Row>
</SimpleTable>
- <SimpleTable name="predictionProbability">
<RowNames />
<ColumnNames>ID;probability-1;probability-2;probability-
3</ColumnNames>

<Row>1;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>2;0.166666666666667;0.666666666666667;0.166666666666667<
/Row>

<Row>3;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>4;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>5;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>6;0.166666666666667;0.666666666666667;0.166666666666667<
/Row>

<Row>7;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>
```

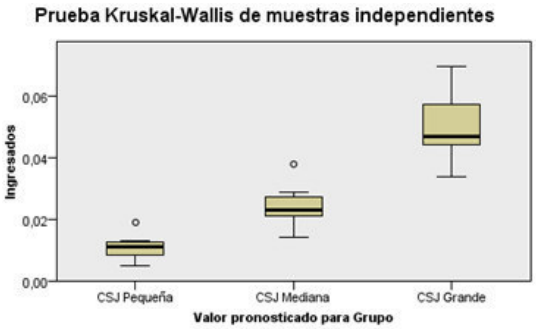
```
<Row>8;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>9;0.666666666666667;0.166666666666667;0.166666666666667<
/Row>

<Row>10;0.166666666666667;0.666666666666667;0.166666666666667
</Row>

<Row>11;0.666666666666667;0.166666666666667;0.166666666666667
</Row>

<Row>12;0.666666666666667;0.166666666666667;0.166666666666667
</Row>

<Row>13;0.666666666666667;0.166666666666667;0.166666666666667
</Row>

<Row>14;0.666666666666667;0.166666666666667;0.166666666666667
</Row>

<Row>15;0.166666666666667;0.166666666666667;0.666666666666667
</Row>

<Row>16;0.166666666666667;0.166666666666667;0.666666666666667
</Row>

<Row>17;0.166666666666667;0.666666666666667;0.166666666666667
</Row>
<Row>18;0.166666666666667;0.5;0.333333333333333</Row>

<Row>19;0.166666666666667;0.666666666666667;0.166666666666667
</Row>

<Row>20;0.166666666666667;0.666666666666667;0.166666666666667
</Row>
<Row>21;0.166666666666667;0.5;0.333333333333333</Row>
<Row>22;0.166666666666667;0.333333333333333;0.5</Row>

<Row>23;0.166666666666667;0.666666666666667;0.166666666666667
</Row>
<Row>24;0.166666666666667;0.5;0.333333333333333</Row>

<Row>25;0.166666666666667;0.666666666666667;0.166666666666667
</Row>

<Row>26;0.166666666666667;0.166666666666667;0.666666666666667
</Row>

<Row>27;0.166666666666667;0.166666666666667;0.666666666666667
</Row>

<Row>28;0.166666666666667;0.166666666666667;0.666666666666667
</Row>
</SimpleTable>
- <ComplexTable name="modelQuality">
```

```
- <SimpleTable name="errorSummary">
  <RowNames />

    <ColumnNames>response;percentIncorrectlyClassifiedCases</ColumnNam
    es>
  <Row>Grupo;0.178571428571429</Row>
    </SimpleTable>
- <SimpleTable name="confusionMatrix">
  <RowNames />
  <ColumnNames>1;2;3</ColumnNames>
  <Row>11;0;0</Row>
  <Row>0;7;1</Row>
  <Row>0;4;5</Row>
    </SimpleTable>
    </ComplexTable>
    </NearestNeighborModel>
    </NearestNeighborModels>
    </Extension>
    </PMML>
```

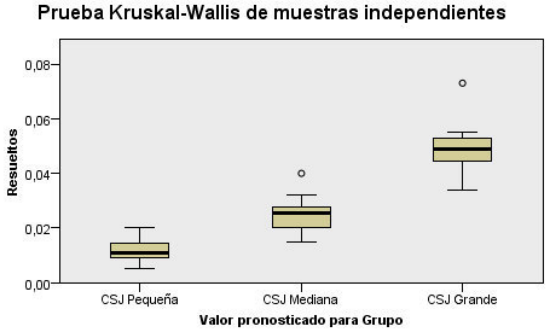
Anexo 8: ***Prueba Kruskal – Wallis de las variables.***



N total	30
Probar estadística	24,127
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

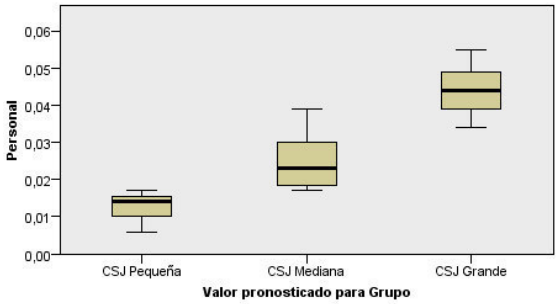


N total	30
Probar estadística	24,868
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000



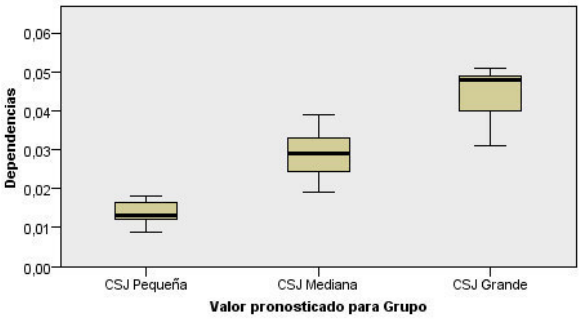
N total	30
Probar estadística	23,918
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba Kruskal-Wallis de muestras independientes



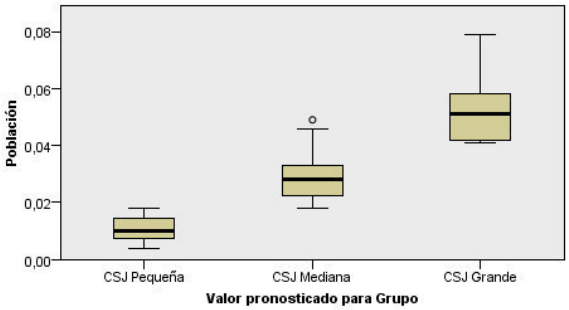
N total	30
Probar estadística	24,027
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba Kruskal-Wallis de muestras independientes



N total	30
Probar estadística	24,168
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

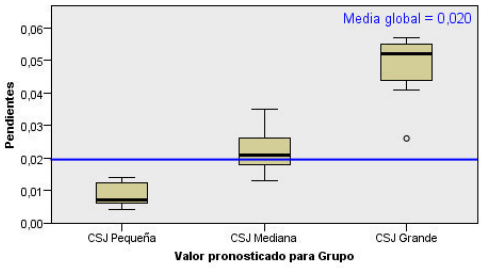
Prueba Kruskal-Wallis de muestras independientes



N total	30
Probar estadística	23,875
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

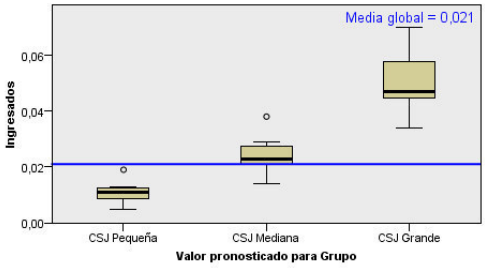
***Anexo 9**: Prueba de medianas para las variables.*

Prueba de medianas de muestras independientes



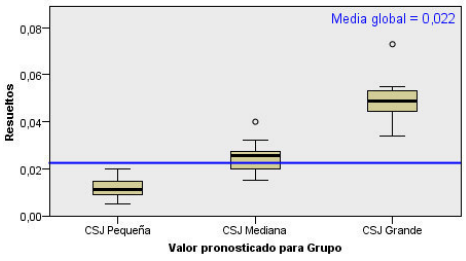
N total	30
Mediana	,020
Probar estadística	19,333
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba de medianas de muestras independientes



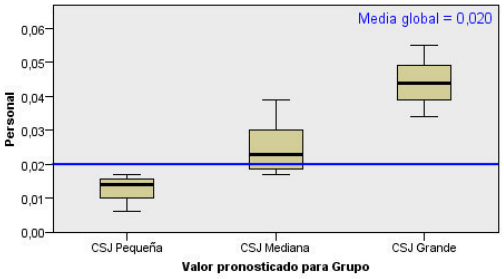
N total	30
Mediana	,021
Probar estadística	18,281
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba de medianas de muestras independientes



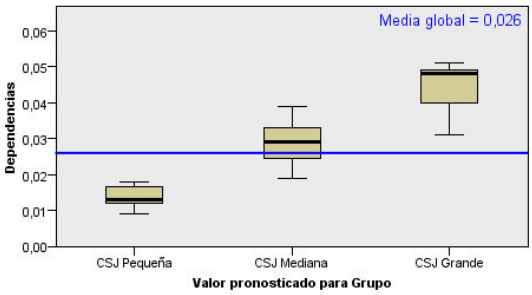
N total	30
Mediana	,022
Probar estadística	19,333
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba de medianas de muestras independientes



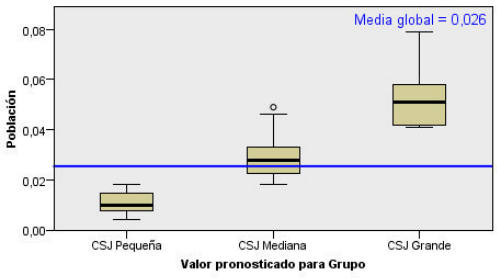
N total	30
Mediana	,020
Probar estadística	19,333
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba de medianas de muestras independientes



N total	30
Mediana	,026
Probar estadística	19,333
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Prueba de medianas de muestras independientes

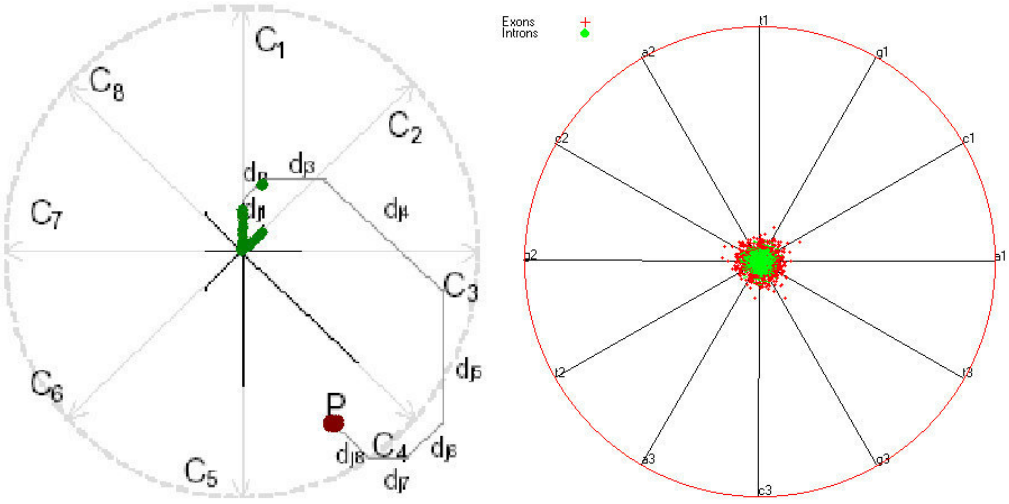


N total	30
Mediana	,026
Probar estadística	19,333
Grados de libertad	2
Sig. asintótica (prueba de dos caras)	,000

Anexo 10: **Gráfico de estrellas.**

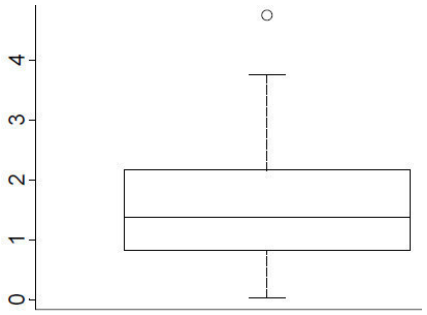
Los gráficos de estrella (Chambers, 1983), asignan a cada unidad de observación una estrella con tantos rayos o ejes (igualmente espaciados y que confluyen en un centro geométrico) como variables queramos representar. Las longitudes de los rayos son proporcionales a los valores de las variables en la observación asociada a la estrella. Los extremos de los rayos se conectan con segmentos de líneas rectas para formar una estrella. De esta forma, podemos agrupar las observaciones según las similitudes que presentan las estrellas. En todas las estrellas se usa siempre el mismo rayo o eje para representar la misma variable. En resumen cada estrella representa una observación, y las variables empiezan a representarse desde la derecha y en dirección a las agujas del reloj. El tamaño de cada línea, respecto al centro de la estrella, está relacionado con los valores re-escalados de cada variable.

- Cada dimensión se presenta como un eje
- Valores en cada dimensión es representado como un vector
- Datos con escalados a la longitud del eje.
  - Valor mínimo cerca al origen
  - Valor máximo al final



Anexo 11: *Boxplot o Caja de Tukey.*

Permite observa de una forma clara la distribución de los datos y sus principales características. Además compara diversos conjuntos de datos simultáneamente. Como herramienta visual se puede utilizar para ilustrar los datos, para estudiar simetría, para estudiar las colas, y supuestos sobre la distribución, también se puede usar para comparar diferentes poblaciones. Se construye de la siguiente manera.



a. Un rectángulo, usualmente orientado con el sistema de coordenadas tal que el eje vertical tiene la misma escala del conjunto de datos.
b. La parte superior y la inferior del rectángulo coinciden con el tercer cuartil y el primer cuartil de los datos.
c. Esta caja se divide con una línea horizontal a nivel de la mediana.
d. Se define un paso (di=1.5*RI)" como 1.5 veces el rango intercuartil, y una línea vertical (un bigote) se extiende desde la mitad de la parte superior de la caja hasta la mayor observación de los datos si se encuentran dentro de un paso. Igual se hace en la parte inferior de la caja.
e. Las observaciones (outliers) que caigan más allá de estas líneas son dibujadas individualmente con "*" aquellos que están entre 1.5 di y 3 di de cada extremo y con "o" a aquellos que están a más de 3 di de cada extremo. Algunos paquetes indican a todos los outliers de la misma forma "o".

La gráfica proporciona información acerca de:

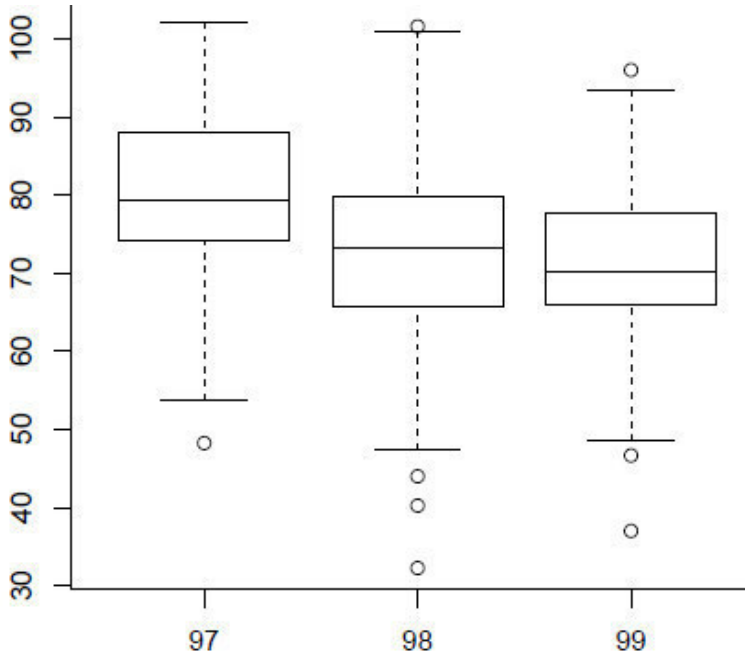
Posición: está representada en la línea que corta la caja y representa la mediana.

Dispersión: está dada por la altura de la caja, como por la distancia entre los extremos de los bigotes.

Sesgo: se observa en la desviación que exista entre la línea de la mediana con relación al centro de la caja y también la relación entre las longitudes de los bigotes.

Las colas: se pueden apreciar por la longitud de los bigotes con relación a la altura de la caja, y también por las observaciones que se marcan explícitamente. La asimetría y los outliers.

Boxplots Paralelos.- Es la comparación de la distribución de dos o más conjuntos de datos graficando en una escala común y en forma paralela los boxplots de cada una de las muestras.



Anexo 12: **Matriz de consistencia**

PROBLEMA	OBJETIVO	HIPÓTESIS
1. Problema General	**1. Objetivo General**	**1. Hipótesis General**
¿Cómo desarrollar modelos mediante el método de los k−vecinos más próximos que permita clasificar y predecir a las 31 Cortes Superiores de Justicia del País?	Encontrar modelos utilizando el método de los k−vecinos más próximos con el propósito de clasificar las 31 Cortes Superiores de Justicia del País y poder realizar predicciones para futuras Cortes Superiores de Justicia.	Los modelos construidos mediante el método de los k−vecinos más próximos son precisos para clasificar las 31 Cortes Superiores de Justicia del País y realizar predicciones futuras.
2. Problemas Específicos	**2. Objetivos Específicos**	**2 Hipótesis Específicos**
¿Cómo implantar un modelo para los predictores (variables) basado en el método de los k vecinos más próximos para clasificar y predecir las Cortes Superiores de Justicia?	Verificar la validez del modelo de clasificación y predicción de las Cortes Superiores de Justicia basado en el método de los k vecinos más próximos.	Los modelos construidos para los predictores (variables) basado en el método de los k vecinos más próximos es eficaz para clasificar y predecir las Cortes Superiores de Justicia.
¿Cómo implementar un modelo de k vecinos más próximos cuando se tiene muestras pequeñas de entrenamiento y reserva (validación)?	Verificar la precisión del modelo de k vecinos más próximo cuando se tiene muestras pequeñas de entrenamiento y reserva (validación)	El modelo de k vecinos más próximos se ejecute con precisión para los datos de las variables, cuando se tiene muestras pequeñas de entrenamiento y reserva.
¿Cómo establecer modelos que permitan identificar y evaluar a las 31 Cortes Superiores de Justicia, respecto de los	Experimentar los modelos que identifican y evalúan a las 31 Cortes Superiores de Justicia, respecto de los predictores	Los modelos descriptivos permiten identificar y evaluar a las 31 Cortes Superiores de Justicia, respecto de los

predictores (variables) en forma *a priori*?	(variables) en forma *a priori*.	predictores (variables) en forma *a priori*.
¿Cómo desarrollar un modelo jerárquico mediante el método de encadenamiento simple (vecinos más próximos) para agrupar las Cortes Superiores de Justicia en conglomerados?	Encontrar un modelo de agrupamiento jerárquico basado en encadenamiento simple (vecinos más próximos) para asociar las Cortes Superiores de Justicia del País en conglomerados.	El modelo de agrupamiento jerárquico mediante encadenamiento simple (vecinos más próximos) permite asociar a las Cortes Superiores de Justicia del País en tres conglomerados.